

# HRT-5 工业机器人控制系统 用 户 手 册

(V1.0)



国家数控系统工程技术研究中心

二〇一四年 五月

## 声明

HRT-5工业机器人控制系统用户手册包括操作说明和编程说明两部分，全面介绍 HRT-5工业机器人控制系统的特性、系统组成、控制命令、操作步骤、用户编程方法及示例等，是用户快速学习和使用工业机器人的基本说明书。

由于篇幅限制及产品开发定位等原因，不能也不可能对系统中所有不必做或不能做的事件进行详细的叙述。因此，本说明书中没有特别描述的事件均可视为“不可能”或“不允许”的事件。

**\* 系统所有规格和设计如有变化，恕不另行通知**

## 目录

声明.....	2
目录.....	3
<b>第一部分 操作篇 .....</b>	<b>1</b>
<b>1 概述.....</b>	<b>2</b>
1.1 HRT-5 工业机器人控制系统简介 .....	2
1.2 工业机器人的典型结构.....	2
<b>2 手动操作 .....</b>	<b>3</b>
2.1 轴的移动.....	4
2.2 机械单元.....	4
2.3 工具坐标系.....	5
2.4 工件坐标系.....	6
2.5 坐标模式.....	6
2.6 校准.....	7
2.7 工具坐标系.....	7
2.8 工件坐标系.....	12
<b>3 示教.....</b>	<b>18</b>
3.1 概述.....	18
3.2 新建程序.....	18
3.3 打开程序.....	19
3.4 程序修改.....	20
3.5 程序检查.....	34
<b>4 自动运行 .....</b>	<b>35</b>
4.1 加载程序.....	35
4.2 自动运行程序.....	36
<b>5 寄存器 .....</b>	<b>37</b>
5.1 R 寄存器.....	37
5.2 位置寄存器.....	37
<b>6 输入/输出 .....</b>	<b>39</b>
6.1 信号显示和设置.....	39
6.2 PLC 备份和恢复 .....	39
<b>7 参数设置 .....</b>	<b>41</b>

7.1	系统参数.....	41
7.2	组参数.....	42
7.3	轴参数.....	43
<b>8</b>	<b>生产管理 .....</b>	<b>45</b>
8.1	概述.....	45
8.2	添加、删除和确认操作人员.....	45
8.3	修改密码.....	46
8.4	程序管理.....	46
8.5	网络设置.....	46
8.6	报警历史.....	47
8.7	操作记录.....	48
	<b>第二部分 编程篇 .....</b>	<b>49</b>
<b>9</b>	<b>运动指令 .....</b>	<b>50</b>
9.1	动作类型.....	50
9.2	位置数据.....	52
9.3	进给速度.....	54
9.4	定位路径.....	55
9.5	附加动作指令.....	56
<b>10</b>	<b>寄存器指令 .....</b>	<b>58</b>
10.1	寄存器 (R) 指令.....	58
10.2	位置寄存器 (PR) 指令.....	59
10.3	位置寄存器轴指令.....	61
<b>11</b>	<b>I/O 指令.....</b>	<b>63</b>
11.1	数字输入/输出 (DI/DO) 指令.....	63
11.2	模拟输入/输出 (AI/AO) 指令.....	65
<b>12</b>	<b>条件指令 .....</b>	<b>148</b>
12.1	寄存器条件比较指令.....	148
12.2	输入输出条件比较指令.....	149
12.3	复合条件的使用.....	150
<b>13</b>	<b>等待指令 .....</b>	<b>151</b>
13.1	指定时间的等待指令.....	151
13.2	条件等待指令.....	151
<b>14</b>	<b>流程控制指令 .....</b>	<b>154</b>
14.1	标签指令.....	154
14.2	程序结束指令.....	154
14.3	无条件跳转指令.....	155

---

14.4	子程序调用指令.....	155
<b>15</b>	<b>其他指令 .....</b>	<b>157</b>
15.1	坐标系指令.....	157
15.2	用户报警指令.....	159
15.3	倍率指令.....	160
15.4	注释指令.....	160
15.5	信息指令.....	161
15.6	预读指令.....	161
<b>16</b>	<b>附录：程序报警定义 .....</b>	<b>163</b>



# 第一部分 操作篇

# 1 概述

## 1.1 HRT-5 工业机器人控制系统简介

HRT-5工业机器人控制系统主要由 HRT 机器人控制器与 HTP 机器人示教器以及运行在这两种设备上的软件所组成。机器人控制器一般安装于机器人电柜内部，控制机器人的伺服驱动、输入输出等主要执行设备；机器人示教器一般通过电缆连接到机器人电柜上，作为上位机通过以太网与控制器进行通讯。借助 HTP 示教器，用户可以实现 HRT-5工业机器人控制系统的主要控制功能：

- 手动控制机器人运动；
- 机器人程序示教编程；
- 机器人程序自动运行；
- 机器人运行状态监视；
- 机器人控制参数设置。

注：本文所说的机器人均指工业机器人，特此说明。

## 1.2 工业机器人的典型结构

用户在操作机器人之前，需要了解工业机器人基本结构与运动形式，并掌握 HRT-5工业机器人控制系统对机器人关节正方向、关节参考点、基坐标系与手腕中心点的定义。

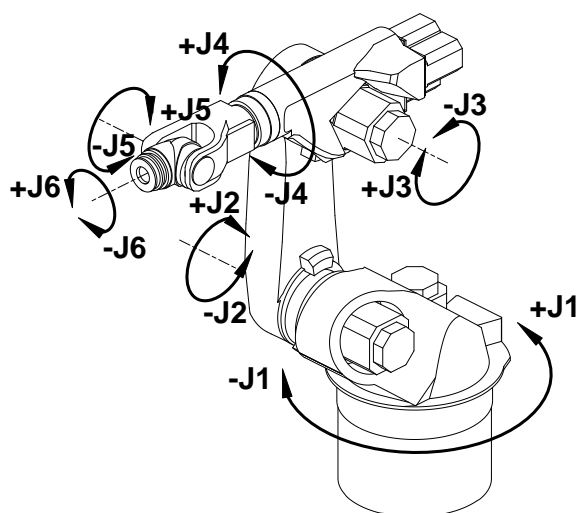


图 1-1 6轴工业机器人关节

图 1-1为一典型的6轴工业机器人，J1、J2、J3为定位关节，机器人手腕的位



置主要由这三个关节决定；J4、J5、J6为定向关节，主要用于改变手腕姿态。HRT-5机器人控制系统对各关节正方向的定义如图1-2所示。可以简单地记为 J2、J3、J5关节以“抬起/后仰”为正，“降下/前倾”为负；J1、J4、J6关节满足“右手定则”，即拇指沿关节轴线指向机器人末端，则其它四指方向为关节正方向。

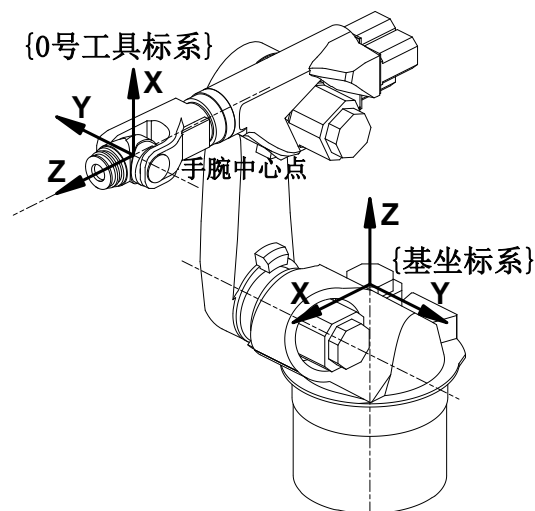


图 1-2 6轴工业机器人参考坐标系

HRT-5工业机器人控制系统采用标准 D-H 法则定义机器人坐标系，即 J1与J2关节轴线的公垂线在 J1轴线上的交点为基坐标系原点，坐标系方向如图 1-2所示：J4、J5、J6关节轴线共同的交点为手腕中心点，0号工具坐标系位于该点，坐标系方向如图 1-2所示。图 1-1和图 1-2所示同时也是关节坐标系的参考点位置，一般机器人在该点设有参考点刻度标识，用于机器人关节轴校准，HRT-5工业机器人控制系统定义在该位置下 J1~J6角度依次为 $0^\circ$ 、 $90^\circ$ 、 $0^\circ$ 、 $0^\circ$ 、 $0^\circ$ 、 $0^\circ$ 。

## 2 手动操作

如图2-1所示，手动操作界面是机器人控制统的主窗口界面，主要用于显示和设置当前组号、运行模式、坐标系等，用户可以在此界面中查看当前的状态信息，并进行设置。

此界面分为两部分，上半部分用于显示坐标位置和控制轴的运转，通过点动模式控制机器人运转。

下半部分用于显示关键信息，点击相应按钮，即可对指定的信息进行设置和操作。

## 2.手动操作



图 2-1 手动操作界面

### 2.1 轴的移动

点击修调值的“+”或“-”调整修调值大小，修调值依次为 VFINE、FINE、1、2、3、4、5、10、20、30、40、50、60、70、80、90、100。其中 VFINE 是增量模式，且步长为1，FINE 也是增量模式，且步长为10。其他值均为连续模式。

连续模式下，点击“J1+”或“J1-”等按钮会使相应的轴移动，长按“J1+”或“J1-”等按钮时，相应的轴会一直移动（当轴移动时，界面上显示的相应坐标值也会随之改变）。

增量模式下，点击“J1+”或“J1-”等按钮会使相应的轴移动指定的步长，不论按下的时间长短。

### 2.2 机械单元

在机械单元界面主要用于设置当前组号，本机器人控制系统可取的组号为1~5。点击“组1”可弹出窗选择组，如图2.2所示：



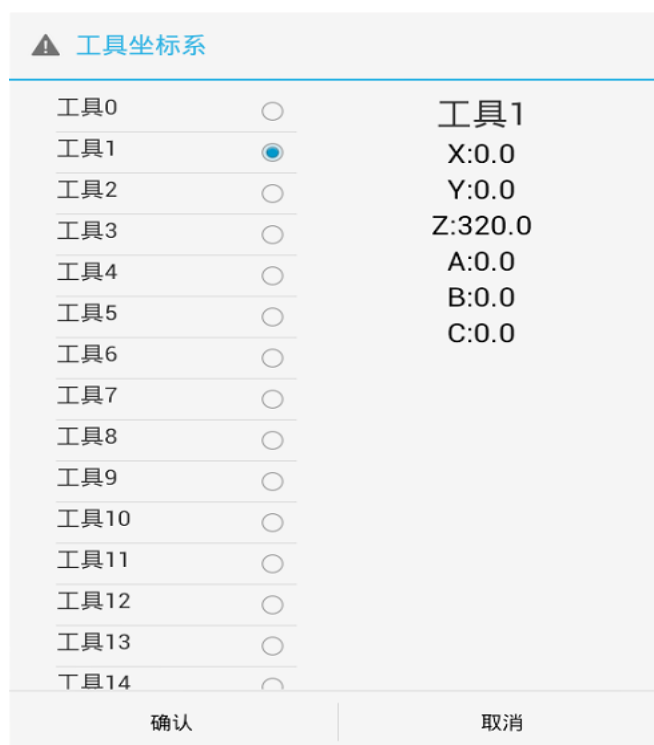
组号选择	
组1	未启用 <input checked="" type="radio"/>
组2	未启用 <input type="radio"/>
组3	未启用 <input type="radio"/>
组4	未启用 <input type="radio"/>
组5	未启用 <input type="radio"/>
确认	取消

图 2-2 机械单元组号设置

如果按下【取消】键，页面回到手动操作界面，组号不修改。

## 2.3 工具坐标系

机器人控制系统共有16个工具坐标系，从工具0到工具15。点击“工具0”可选择相应工具坐标系，当选中一个工具坐标系时，右侧会显示该坐标系的坐标值供用户参考。如图2.3所示：



工具坐标系		
工具0	<input type="radio"/>	工具1
工具1	<input checked="" type="radio"/>	X:0.0
工具2	<input type="radio"/>	Y:0.0
工具3	<input type="radio"/>	Z:320.0
工具4	<input type="radio"/>	A:0.0
工具5	<input type="radio"/>	B:0.0
工具6	<input type="radio"/>	C:0.0
工具7	<input type="radio"/>	
工具8	<input type="radio"/>	
工具9	<input type="radio"/>	
工具10	<input type="radio"/>	
工具11	<input type="radio"/>	
工具12	<input type="radio"/>	
工具13	<input type="radio"/>	
工具14	<input type="radio"/>	
确认		取消

图 2-3 工具坐标系设置窗口

### 2.4 工件坐标系

机器人控制系统共有16个工件坐标系，从工件0到工件15。点击“工件0”可选择相应工件坐标系，当选中一个工件坐标系时，右侧会显示该坐标系的坐标值供用户参考。如图2.4所示：

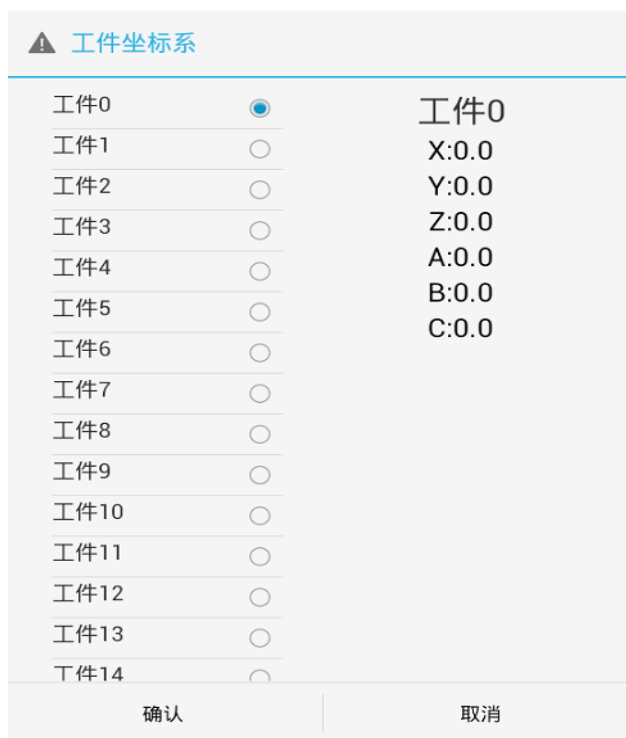


图 2-4 工件坐标系设置窗口

### 2.5 坐标模式

机器人控制系统有五种坐标模式，分别为关节坐标、基坐标、工具坐标、工件坐标。

在附加变位机等外部设备时，还可以使用外部轴坐标。

点击“关节坐标”可选择坐标类型，如图2-5所示。其余类推。



图 2-5 坐标模式设置

## 2.6 校准

对机器人的六个（或更多）关节轴进行校准：

首先，在手动模式下控制机器人各关节轴移动至标准零点姿态；然后，在下图的校准界面中输入各关节轴的零点值（如轴一到轴六分别为 0,90,0,0, -90,0 或者 0,90,90,0, -90,0）；最后，按下确认键，完成校准。

注：校准操作完成后，系统可能提示“重启后生效”，请重启控制系统。

对轴进行校准。点击“校准”，可调整机器人各轴的运动误差，如图 2-6 所示。选中需要校准的轴，输入校准值，即可校准该轴。



校准

轴号	关节轴
轴1	***
轴2	***
轴3	***
轴4	***
轴5	***
轴6	***

确认 取消

图 2-6 校准界面

## 2.7 工具坐标系

机器人控制系统支持16个工具坐标系设定。点击“工具坐标设定”，可设置相应工具坐标系或工件坐标系的各个坐标值，如图2-7所示：



图 2-7 工具坐标系设定界面

在图 2.7 所示对话框中，选择一个工具坐标系，右侧将显示选中坐标系的坐标值。点击一个轴的坐标值，即可进行坐标值的修改。点击“清除坐标”，可将当前选中的坐标系的各个坐标值清零。

### 2.7.1 三点标定

通过标定空间中机器人末端在坐标系中的三个不同位置来计算工具坐标系。

工具坐标系三点标定操作步骤如下：

- (1) 点击工具坐标系进入到工具坐标系界面，选中需要标定的工具号（工具0不能被标定），点击“坐标标定”，可弹出坐标标定对话框，如图2-8所示：



图 2-8 坐标标定

- (2) 在图2.8所示对话框中，点击“三点标定”弹出下拉框可选择标定方式，此处选择三点标定，选中一个被标定点，如选择“接近点1”，然后点击“修改位置”，会显示图2.9所示手动运行界面，供用户手动修改该点位

置。



图 2-9 修改位置

- (3) 点动机器人到你想要记录的点，三个点的记录原则请参考 TCP 三点法标定说明（2.8 工件坐标系中的三点法），点击“记录位置”即为确认修改接近点的坐标值，接近点1位置记录完成，如图2-10所示：



图 2-10 接近点 1 指定完成

- (4) 按照上述方法指定接近点2、3的位置，当三个位置都显示已修改时，按下“确认”，即完成三点标定，此时相应工具标定完成。

### 2.7.2 六点标定

通过标定空间中机器人末端六个不同姿态的位置来计算工具坐标系。

工具坐标系六点标定的详细操作步骤如下：

- (1) 点击工具坐标系进入工具坐标系界面，选中需要标定的工具号（工具0不能被标定），点击“坐标标定”，弹出坐标标定对话框，如图2-11所示：



图 2-11 六点标定

- (2) 在图2-11所示对话框中，点击“三点标定”弹出下拉框可选择标定方式，此处选择六点标定，选中一个被标定点，如选择“接近点1”，然后点击“修改位置”，则会显示图2-12所示手动运行界面，供用户手动修改该点位置。





图 2-12 修改位置

- (3) 点动机器人到你想要记录的点，六个点的记录原则请参考 TCP 六点法标定说明。点击“记录位置”即为确认修改接近点的坐标值，接近点1位置记录完成。



图 2-13 接近点 1 指定完成

## 2.手动操作

- (4) 按照上述方法指定其余点的位置，当六个位置都显示“已修改”时，按下“确认”，即完成六点标定，此时相应工具标定完成。

### 2.8 工件坐标系

工件坐标系是由用户在工件空间定义的一个笛卡尔坐标系。工件坐标包括： $(X,Y,Z)$  用来表示距原点的位置， $(A,B,C)$  用来表示绕 X-，Y-，Z-轴旋转的角度。与工具坐标系相同，机器人控制系统支持16个工件坐标系设定，每个工件坐标系可以属于不同的组号，也可为每个工件坐标系添加相应的注释说明。

依次操作[**I**] → [设置] → [工件坐标系]，即可进入工件坐标系界面，如图2-14所示，右边窗口上部分显示所有工件坐标系，下窗口显示当前选中工件号的坐标值。当上窗口中的工件号变动时，下窗口的坐标值也随之变化。



图 2-14 工件坐标系设定

在图 2-14 所示对话框中，选择一个工具坐标系，右侧将显示选中坐标系的坐标值。点击一个轴的坐标值，即可弹出窗口修改选中的坐标值。点击“清除坐标”，可将当前选中的坐标系的各个坐标值清零。

工件坐标系可以用以下两种方式进行标定：

#### 1. 三点法

将第一个标定点定为工件坐标系 X 轴的起点（如图2-15所示），将工具 TCP

(即工具坐标中心点)沿工件坐标系+X方向移动一定距离作为X方向延伸点,再从工件坐标系XOY平面第一或第二象限内选取任意点作为Y方向延伸点。第一个标定点为工件坐标系绝对原点。由此三个点计算出工件坐标系。

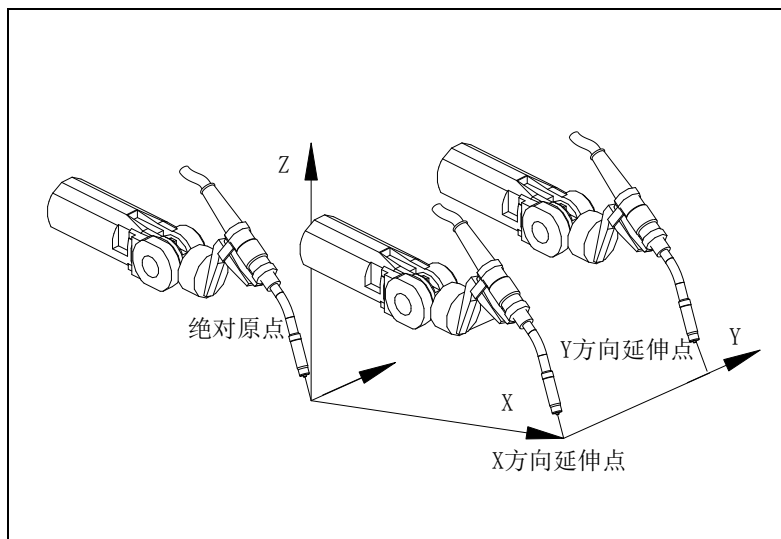


图 2-15 工件三点法标定

## 2. 四点法

将第一个点作为相对原点,将工具TCP沿工件坐标系+X方向移动一定距离作为X方向延伸点,再从工件坐标系XOY平面第一或第二象限内选取任意点作为Y方向延伸点,最后操作机器人到第四个点,作为绝对原点。由此四个点计算出工件坐标系。

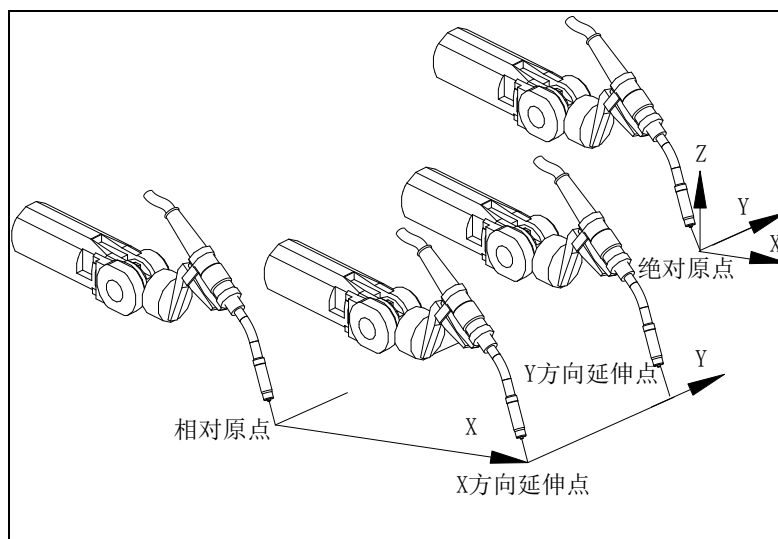


图 2-16 工件四点法标定

### 2.8.1 三点标定

工具坐标系三点标定的详细操作步骤如下：

- (1) 首先，点击工具坐标系进入到工具坐标系界面，选中需要标定的工具号（工具0不能被标定），点击“坐标标定”，可弹出坐标标定对话框，如图2-17所示：



图 2-17 坐标标定

- (2) 在图2-17所示对话框中，点击下拉框可选择标定方式，此处选择三点标定，选中一个被标定点，如选择“接近点1”，然后点击“修改位置”，会显示图2-18所示手动运行界面，供用户手动修改该点位置。



图 2-18 修改位置

- (3) 点动机器人到你想要记录的点，三个点的记录原则请参考 TCP 三点法标定说明，点击“记录位置”即为确认修改接近点的坐标值，接近点1位置记录完成。界面如图2-19所示，开始记录接近点2的位置：



图 2-19 接近点 1 指定完成

- (4) 按照上述方法指定接近点2、3的位置，当三个位置都显示已修改时，按下“确认”，即完成三点标定，此时相应工具标定完成。

## 2.8.2 四点标定

工具坐标系六点标定的详细操作步骤如下：

- (1) 首先，点击工具坐标系进入到工具坐标系界面，选中需要标定的工具号（工具0不能被标定），点击“坐标标定”，可弹出坐标标定对话框，如下所示。

## 2.手动操作



图 2-20 六点标定

- (2) 在图2-20所示对话框中，点击下拉框可选择标定方式，此处选择六点标定，选中一个被标定点，如选择“接近点1”，然后点击“修改位置”，会显示图2-21所示手动运行界面，供用户手动修改该点位置。



图 2-21 修改位置

- (3) 点动机器人到你想要记录的点，六个点的记录原则请参考 TCP 六点法标定说明。点击“记录位置”即为确认修改接近点的坐标值，接近点1位置记录完成。



图 2-22 接近点 1 指定完成

- (4) 按照上述方法指定其余点的位置，当六个位置都显示已修改时，按下“确认”，即完成六点标定，此时相应工具标定完成。

# 3 示教

本章介绍一些示教界面中指令的基本操作方法，详细的编程过程请参照后面的指令编程章节。

## 3.1 概述

程序的基本信息包括：程序名、程序注释、子类型、组标志、写保护、程序指令和程序结束标志。

(1) 程序名：用以识别存入控制器内存中的程序，在同一个目录下不能包含两个或更多拥有相同程序名的程序。程序名长度不超过8个字符，由字母、数字、下划线(\_)组成。

(2) 程序注释：程序注释连同程序名一起来描述选择界面上显示的附加信息。与程序名相关（非程序内容注释），最长16个字符，由字母、数字及符号（\_、@、\*）组成。新建程序之后可在程序选择中修改程序注释。

(3) 子类型：用于设置程序文件的类型。目前本系统只支持机器人程序这一类型。

(4) 组标志：设置程序操作的动作组，必须在程序执行前设置。目前本系统只有一个操作组，默认的操作组是组1（1，\*，\*，\*，\*）。

(5) 写保护：指定该程序可否被修改。若设置为“是”，程序名、注释、子类型、组标志等不可修改。若此项设置为“否”，程序信息可修改。当程序创建且操作确定后，可将此项设置为“是”来保护程序，防止他人或自己误修改。

(6) 程序指令：包括运动指令、寄存器指令等示教中涉及的所有指令。

(7) 程序结束标志：程序结束标志（END）自动在程序的最后一条指令的下一行显示。只要有新的指令添加到程序中，程序结束标志会在屏幕上向下移动，所以，它总在最后一行。当执行完最后一条指令后，程序执行到程序结束标志时，就会自动返回到程序的第一行并终止。

## 3.2 新建程序

点击示教界面下方左侧的“新建程序”按钮，在弹出的对话框中输入程序名，可新建一个空的程序文件，如图 3-1 所示。



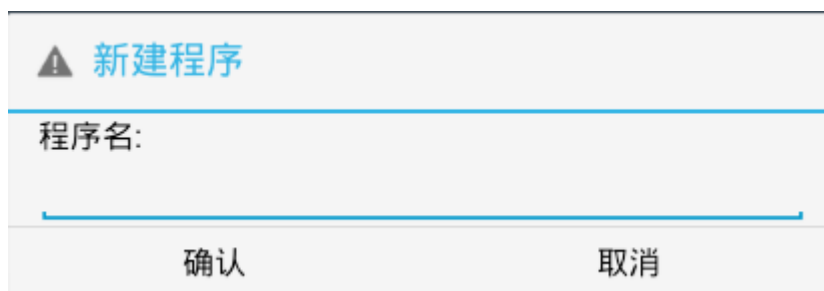


图 3-1 程序名输入对话框



图 3-2 新建程序的示教窗口

### 3.3 打开程序

打开程序对话框可查看系统中所有的程序文件及其属性，点击“打开程序”，可显示图3-3所示程序文件列表，选择一个现有的程序文件并点击“确认”后可加载该选中的程序文件。

### 3.示教



图 3-3 打开程序窗口

## 3.4 程序修改

示教主要提供程序修改编辑功能，对于触摸屏手持器，本机器人控制系统提供两种操作方式即：短按和长按。

### 3.4.1 行内编辑（短按）

短按任一行（即点击）的程序语句（最后一行“END”除外），可对该行程序语句的内容进行编辑。

以程序行是运动指令为例，如图3.4所示：



图 3-4 行编辑弹出框

本行编辑界面设计如下：



图 3-5 本行编辑


- (1) 当前指令行：显示当前编辑的指令行，其中蓝色显示的是当前的编辑项；
- (2) 指令帮助说明：点击帮助图标  后，显示指令类型说明，运动指令时，弹出帮助对话框如图3.6所示：



图 3-6 示教帮助说明

- (3) 可滑动的指令编辑行：蓝色高亮显示部分即为当前的编辑项，可左右滑动，以选择当前的编辑项；
- (4) 当前编辑项的可选项：根据当前编辑项，显示当前可选择的选项，如“J”指令的可选项为“J”、“L”和“C”；
- (5) 本次编辑的确认和取消：点击“确认”后，即可将本次编辑好的指令行替换旧的编辑行。或点击“取消”按钮，取消当前编辑。

### 3.示教

对指令行中的数据进行编辑时，界面将显示如图3-7所示：



图 3-7 IO 指令行内编辑界面

点击输入框，输入数据后点击“确认”，即可完成数据的编辑。

#### 3.4.2 行编辑（长按）

长按任一行的程序语句，可对该行程序语句做整体操作，包括删除、复制、剪切、粘贴、修改位置、上行插入、下行插入等，如图3-8所示：



图 3-8 程序编辑弹出框

## 1) 删除

删除当前选择的行。

## 2) 复制

复制当前选择的行内容到粘贴板。

## 3) 剪切

复制当前选择的行内容到粘贴板，并删除当前行；

## 4) 粘贴

当前行后移，将粘贴板上的行信息粘贴为当前行；

## 5) 位置修改

若当前行含有位置变量 **P** 或者位置寄存器 **PR**，且位置号都是直接寻址的（即为“**P[常量]**”或“**PR[常量]**”），长按当前行时，修改位置菜单颜色变为可操作，即可以对当前行的位置信息进行查看或修改。

如本例位置号为常量“8”，即可对位置信息进行查看和修改，在弹出的操作菜单中选择“修改位置”，弹出如图3-9所示对话框：

### 3.示教



图 3-9

对话框左边进行模式选择，右边显示坐标及坐标值的修改。

可直接点击坐标值进行修改，也可点击“位置修改”进入到手动界面进行位置修改，将机器人移动到所需要的位置，如图3-10所示：

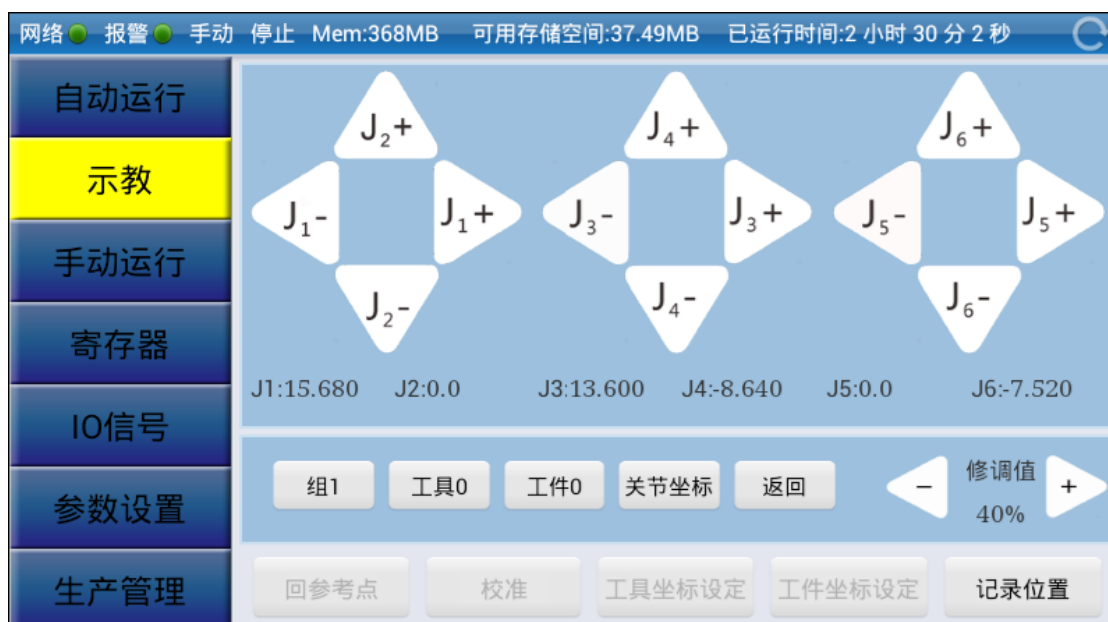


图 3-10 位置修改界面

点击“记录位置”按钮，即可返回到位置变量修改对话框，此时坐标值已修改，已记录下机器人当前位置，如图3-11所示：

▲ 位置变量设置			
当前选择: P[8]			
模式选择:		坐标值:	
机械单元:	组1	J1	15.680 deg
坐标类型:	直角 关节	J2	0.000 deg
		J3	13.600 deg
		J4	-8.640 deg
		J5	0.000 deg
		J6	-7.520 deg
		位置修改	
确认		取消	

图 3-11 位置修改完成后界面

点击“确认”按钮，即可完成位置修改。或点击“取消”按钮，取消当前修改。

## 6) 插入（上行或下行）

根据指令行插入的位置，插入操作分为上行插入和下行插入。

- 上行插入：是在当前所选择指令的前一行插入指令；
- 下行插入：是在当前所选择指令的后一行插入指令。

下面以上行插入为例，介绍插入指令行的操作方法。

- (1) 选择“上行插入”选项，弹出如图3-12所示指令选择框：

### 3.示教



图 3-12 指令类型选择界面

(2) 选择指令类型，如选择“I/O 指令”，界面切换至 IO 指令的编辑对话框；



图 3-13 I/O 指令编辑对话框

(3) 该编辑界面即为行内编辑的界面，编辑完成后点击确认即可完成上行插入操作。

## 7) 编辑本行

可对本行程序进行编辑，如同短按程序行进行行内编辑效果是一样的。



### 3.4.3 指令类型说明

本小节简单介绍不同类型的指令，详细编程方法请参看编程篇。

#### 1) 运动指令

运动指令类型包括三种：关节定位（J）、直线定位（L）、圆弧定位（C）。



图 3-14 运动指令

#### 2) 寄存器指令

寄存器指令主要是在寄存器上完成算术运算。根据运算表达式左值的类型，可以将寄存器指令分为：R 寄存器、位置寄存器 PR[i]指令及位置寄存器轴指令 PR[i,j]。

寄存器指令支持的运算操作有：+、-、\*、/、MOD（取商的余数，即小数点后的值）、DIV（取商的整数）。如图3-15所示，但对于位置寄存器只支持+、-两种运算操作。寄存器指令的详细编程说明，请参看后面章节的编程说明。



图 3-15 寄存器指令

寄存器指令支持多项式运算，但使用时请注意：单独一行最多5个运算符。

本系统支持+、-、\*、/、MOD、DIV 在一行中混合使用，MOD、DIV 属于同一级别，\*、/属于同一级别，+、-属于同一级别，优先级别划分为：

- MOD、DIV优先级高于\*、/；
- \*、/优先级高于+、-；
- 同一级别下，优先级从左到右，即左边高于右边；

在实际的示教编程中，如果想要取消后面的多项式，如：

示例1：  $R[2]=R[3]-R[4]+R[5]-R[6]$

示例1中，若要取消“+R[5]-R[6]”，选中“+”运算，在运算符（如+、-、\*、/都是运算符）的输入选择窗口中选择“<cr>”，即可清除选中运算符后面的所有表达式即“+R[5]-R[6]”。

**注：**对于使用工件（或工具）坐标系作为**右值**的位置寄存器 PR 指令，只能进行直接赋值运算，不能进行+、-、\*、/等多项式运算。

### 3) I/O 指令

I/O 指令用于数字输入/输出(DI/DO), 或模拟量输入/输出 (AI/AO)。主要是一些赋值语句, 如错误!未找到引用源。16所示:



图 3-16 IO 指令

### 4) 条件指令

条件指令由 IF 开头, 用于比较判断是否满足条件, 若满足则执行后面的 JMP 或 CALL 指令。支持的比较运算符有: >、>=、=、<=、<、<>, 如图3-17所示, 还可以使用逻辑与 (AND) 和逻辑或 (OR) 指令对这些条件语句进行运算。



图 3-17 条件指令

在条件指令使用时，请注意以下几点：

- 对于寄存器（R）、模拟量输入/输出及组输入/输出比较指令，可使用全部比较符，但数字量的输入/输出比较时，只能使用=和<>；
- 可使用逻辑与（AND）和逻辑或（OR）指令，每行中允许最多使用5个，但不允许同时使用AND 和OR。

## 5) 等待指令

机器人控制系统的等待指令包括两种：指定时间等待指令和条件等待指令。表达形式如图3-18所示，“WAIT ...sec”为指定时间等待指令，其余的是条件等待指令。



图 3-18 等待指令

**注：**条件等待指令中的条件比较项也可进行 AND、OR 运算，添加和设置的操作方法与条件指令相同。

## 6) 流程控制指令

机器人控制系统支持三种流程控制指令，分别为标签指令（LBL）、程序结束指令（END）和无条件分支指令。其中无条件分支指令包括跳转指令（JMP LBL）和程序调用指令（CALL）。

程序结束指令（END）是在程序创建初期就已默认添加的指令，表示程序已经结束，用户无需手动添加。在菜单中选择“流程控制”，示教界面显示流程控制指令的选择窗口，如图3-19所示：



图 3-19 流程控制指令

## 7) 其他指令

机器人控制系统支持的其他指令包括：坐标系指令、用户报警指令、信息指令、注释指令、倍率指令和预读指令。在菜单树中选择“其他指令”，示教界面将显示其他指令的选择窗口，如图3-20所示：



图 3-20 其他指令

### 1) 坐标系指令

坐标系指令包括设置和选择两种：

- 坐标系设置：UTOOL[i]= ...，UFRAME[i]= ...，其中“i”为常量或寄存器R，“...”为位置寄存器“PR”；
- 坐标系选择：UTOOL\_NUM= ...，UFRAME\_NUM=...，其中“...”为常量或寄存器R。

### 2) 用户报警指令

UALM[i]：显示警报信息，并中止程序的运行。报警信息在界面设置，i指定报警信息的序号（从9000到9099），可为常量或寄存器R。

**注：**若为直接寻址，即输入数据为报警号时，其取值范围为9000到9099，不在此范围输入无效。若为间接寻址，即输入数据为寄存器号，其取值范围为0到199，不在此范围输入无效。

### 3) 信息指令

信息指令用于在界面上弹出提示信息，格式为：MESSAGE [...], 表示在界面上显示“...”指定的信息；

**注：**信息内容的长度限制为24个字符（包括字母、符号和数字），超出这个长度，输入无效。

### 4) 注释指令

由注释符“!”指定。表示“!”后面的字符为注释，不参入程序运行。在指令选择窗口中为“COMMENT”，在程序指令中表现为“!”。

**注：**信息内容的长度限制为32个字符（包括字母、符号和数字），超出这个长度，输入无效。

### 5) 倍率指令

倍率指令用于改变进给速度倍率，其格式为：OVERRIDE=...，其中“...”可为常量或寄存器R，倍率的范围是1%-100%。

**注：**若为直接寻址，即输入数据为倍率值时，其取值范围为1到200，不在此范围输入无效。若为间接寻址，即输入数据为寄存器号，其取值范围为0到199，在此范围输入无效。

### 3.5 程序检查

系统支持对编写的程序进行语法检查，若程序有语法错误，提示报警号、出错程序及错误行号。

程序报警定义请参照本手册后面的附录，错误提示信息中括号中的数据即为报警号。若程序没有错误，提示程序检查完成。

建议用户在首次运行新编写的程序之前，先执行程序检查，以保证程序的正常运行。



## 4 自动运行

在自动操作模式下可以运行机器人程序，任何程序都必须先加载到内存中才能运行。自动运行界面如图4-1所示：



图 4-1 自动运行界面

### 4.1 加载程序

用户可选择并加载现有的程序文件，启动加载了的程序后，机器人会根据程序文件的内容进行相关的动作。

点击“加载程序”，会显示当前可用的所有程序文件的列表，如图4-2所示，选择所需加载程序文件并点击“确认”即可加载选定程序文件。

## 4.自动运行



图 4-2 程序加载界面

### 4.2 自动运行程序

该界面中的“启动/暂停”按钮和“停止”按钮可控制程序运行的启停。

“连续/单步”按钮和“单周/循环”按钮可设置程序自动运行的方式。

选择单步运行模式，系统会在运行完一程序后停止；若为连续运行，则系统连续运行完程序。选择单周运行模式，系统会在运行完当前程序后停止；若为循环运行，则系统运行完程序后，再次从程序首行重新运行。

该界面中的“关节/直角”按钮可切换程序运行时，不同类型坐标系下各轴的当前坐标值。

点击修调值修改按钮“+”和“-”可修改程序运行时的修调值大小。

## 5 寄存器

点击左边导航中的“寄存器”即可进入 R 寄存器和位置寄存器显示和设置界面。如图5-1所示：

网络 ● 报警 ● 自动 停止 Mem:378MB 可用存储空间:37.38MB 已运行时间:49 秒			
自动运行	R寄存器		位置寄存器
	R[0]	0.0	PR[0] 关节坐标
示教	R[1]	0.0	PR[1] 关节坐标
	R[2]	0.0	PR[2] 关节坐标
手动运行	R[3]	0.0	PR[3] 关节坐标
	R[4]	0.0	PR[4] 关节坐标
寄存器	R[5]	0.0	PR[5] 关节坐标
	R[6]	0.0	PR[6] 关节坐标
IO信号	R[7]	0.0	PR[7] 关节坐标
	R[8]	0.0	PR[8] 关节坐标
参数设置	R[9]	0.0	PR[9] 关节坐标
	R[10]	0.0	PR[10] 关节坐标
生产管理	R[11]	0.0	PR[11] 关节坐标

图 5-1 寄存器界面显示

### 5.1 R 寄存器

控制系统支持200个 R 寄存器变量，寄存器从0开始编号，可以设置 R 寄存器的值。

点击 R 寄存器列表中任一行，可弹出窗口修改所选 R 寄存器的值，如图5-2所示，在确认修改后会自动刷新列表。

⚠
R寄存器设置

当前选择：R[3]

参数：

确认
取消

图 5-2 R 寄存器设置对话框

### 5.2 位置寄存器

位置寄存器作为全局变量，用于存放位置信息。机器人控制系统支持100个位置寄

## 5.寄存器设置

寄存器，寄存器号从0开始编号。支持对指定位置寄存器的坐标类型、组和坐标值进行设置修改。

点击位置寄存器列表中任一行，可弹窗修改所选位置寄存器的各个参数，如图 5-3 所示，在确认修改后会自动刷新该位置寄存器列表。



位置寄存器设置

当前选择: PR[3]

模式选择:	坐标值:
机械单元: 组1	J1 0.0 deg
坐标类型: 直角 关节	J2 0.0 deg
	J3 0.0 deg
	J4 0.0 deg
	J5 0.0 deg
	J6 0.0 deg

位置修改

确认 取消

图 5-3 位置寄存器设置对话框

在图5-3所示的位置寄存器设置对话框中，点击坐标轴值，可以直接对坐标进行设置，也可以点击“修改位置”，切换至手动运行界面，手动修改各个轴的坐标值并点击“记录位置”，即可返回该对话框并显示修改后的各个轴的坐标值。

## 6 输入/输出

机器人控制系统提供了完备的 I/O 通信接口，可以方便地与周边设备进行通信。本系统的 I/O 板提供的常用信号处理有输入信号 X 和输出信号 Y。输入/输出主要是对这些输入/输出状态进行管理和设置。

输入/输出点数及状态从下位机获取。

### 6.1 信号显示和设置

进入该界面时，软件会自动刷新信号状态并显示。

IO 信号的界面显示如下所示：



图 6-1 IO 信号界面

对一个信号进行设置时，要先解开相应的“锁”，即点击相应的 **关** 按钮，开锁后即可修改相应信号的状态。

### 6.2 PLC 备份和恢复

1、用户可将控制系统运行所需的 PLC 程序备份到示教器中，方便存取。点击“PLC 备份”，弹出 PLC 备份对话框，提供备份功能，如图 6.2 所示：



图 6-2 PLC 备份

2、从示教器将已备份 PLC 文件下传至控制系统并重新加载。点击“PLC 恢复”，弹出 PLC 恢复对话框，提供恢复功能。



图 6-3 PLC 恢复

恢复完成后，下位机即时生效。

## 7 参数设置

实现对机器人控制系统的设置，包括权限、参数、工具/工件坐标系及系统日期等。其中参数包括系统参数、组参数和轴参数。本系统支持5个控制组，最多32个物理轴。组参数与轴参数相互关联，每个组最多可以配置9个逻辑轴。用户可根据需要设置物理轴与逻辑轴之间的映射关系，详见组参数设置详细信息。

注意，每个物理轴只能对应一个组的一个逻辑轴，不能进行多重映射。配置好的物理轴可以在轴参数列表中查看该轴所属控制组情况。

参数界面显示如下：



图 7-1 参数设置界面

### 7.1 系统参数

目前，本机器人控制系统支持的系统参数共三个：插补周期、硬件通讯方式（中断或扫描）、报警履历最大记录数及 WAIT 指令 TIMEOUT 时间。

点击进入系统参数界面，如图7-2所示：

## 7.设置



图 7-2 系统参数界面

点击相应的行即可对选中的参数进行设置。

### 注意:

- 1) 第一次进入参数界面时，点击参数列表中任一行，弹出密码输入框，如图 7-3 所示，密码输入正确后才可进入子列表。（初始密码为“003520”）

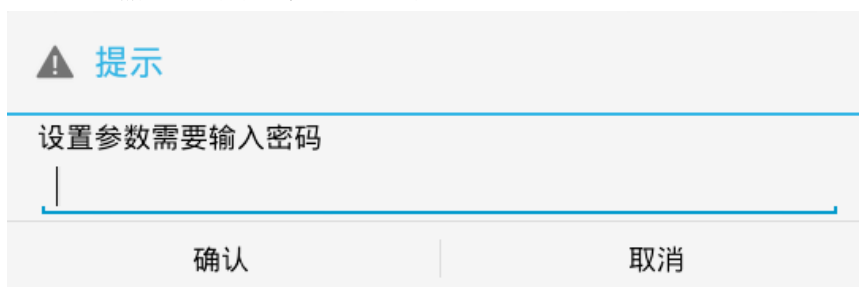


图 7-3 密码输入框

- 2) 参数值是从下位机同步上来的，如果上下位机网络并未连接时，参数值为0或空。

## 7.2 组参数

目前，本机器人控制系统支持5个组号。点击“组参数”进入到组参数界面，如图 7-4所示：





图 7-4 组号选择界面

每个组都拥有各自的组参数集，可分别对其进行设置。点击待设置的“组号”，如“组1”，进入到组号设置界面，如下所示：



图 7-5 组参数设置界面

此时，就可以对指定组号的参数集进行设置。

**注：**对轴号进行设置时，由于每个物理轴只能对应单个组的一个逻辑轴，不能进行多重映射，所以如果有某个组已使用了某个轴号，其他组就不能再使用了。

### 7.3 轴参数

目前，本机器人控制系统支持的轴数共有32个。每个轴可以属于不同的组号。轴参数修改步骤如下：

## 7.设置

(1) 点击“轴参数”进入到轴参数界面，如图7-6所示：

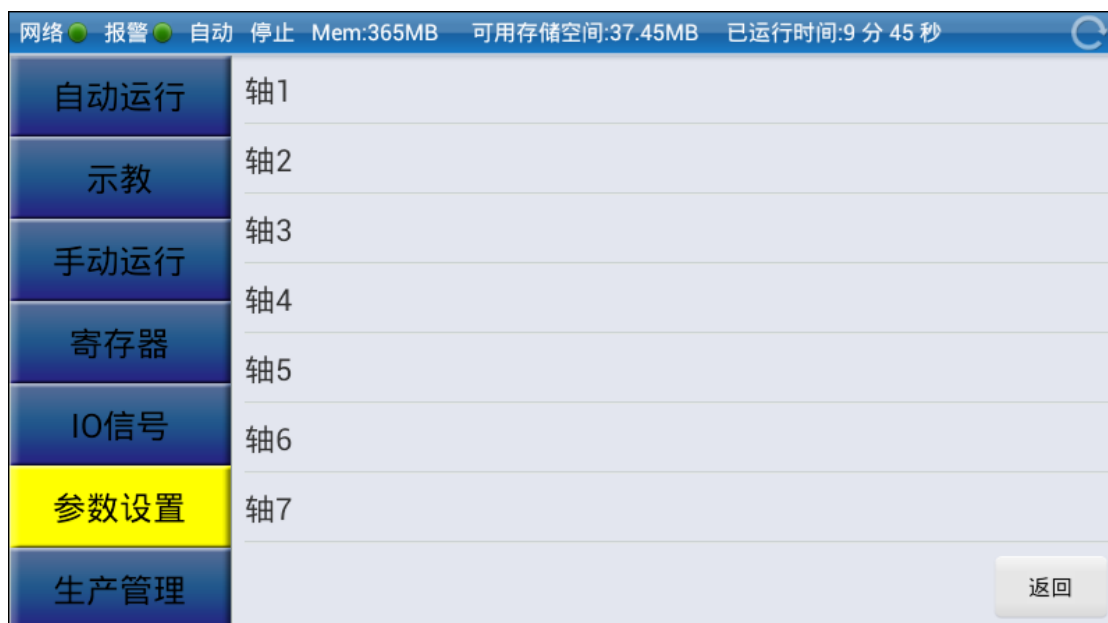


图 7-6 轴参数选择界面

(2) 点击需要设置的轴号，即可进入到选定轴的详细信息显示界面，如选择“轴1”，其详细信息界面如图7-7所示：



图 7-7 轴参数设置界面

(3) 选择需设置的参数行，在弹出的输入框，输入内容后“确认”即可完成参数数值设置。

## 8 生产管理

生产管理的显示和操作界面如图 8-1 所示。

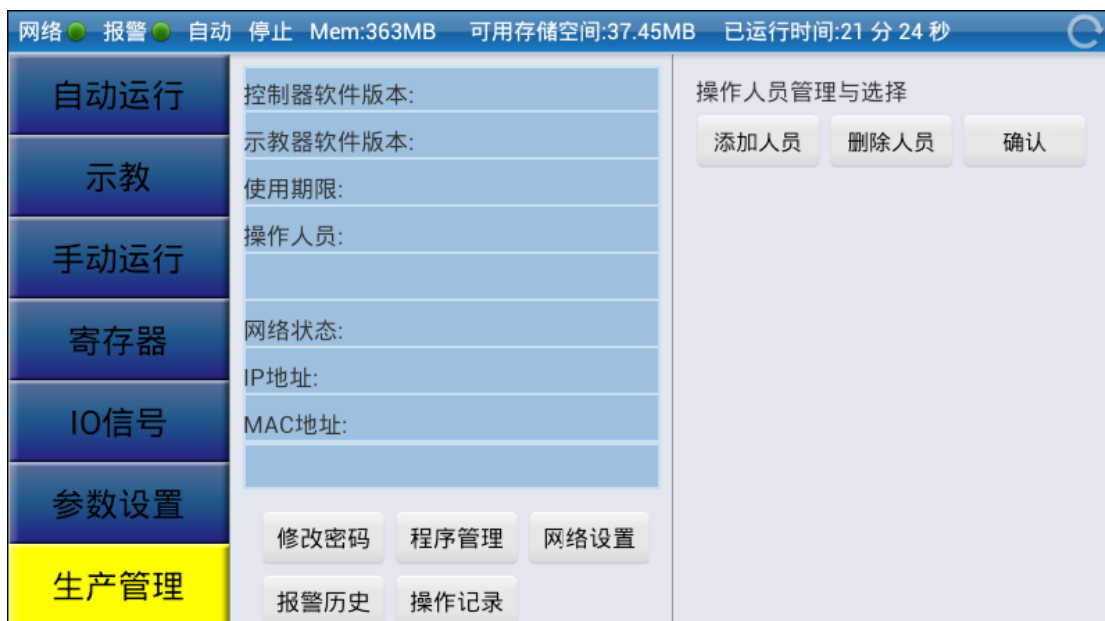


图 8-1 生产管理界面

### 8.1 概述

该界面主要显示与生产相关的一些信息，如软件版本、使用期限、操作人员、以及当前连接的网络状态，报警历史和操作记录。

### 8.2 添加、删除和确认操作人员

点击右边的“添加人员”会弹出如图 8-2 所示对话框，输入工号和姓名，点击确认即可添加操作人员。

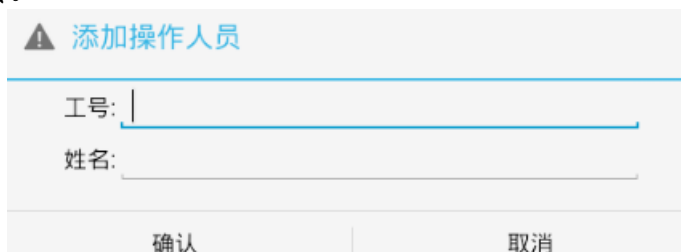


图 8-2 添加操作人员

选中你想要删除的操作人员，点击“删除人员”，弹出对话框,点击确认即可删除该选定操作人员。

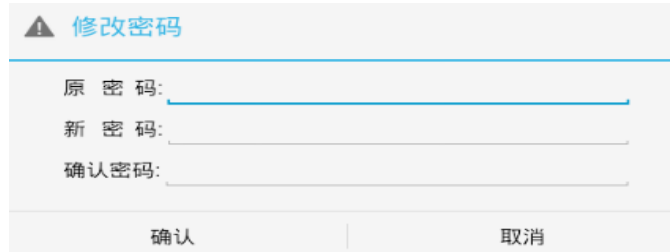
选中你想要确认的操作人员，点击“确认”，该操作人员就被选为当前的操作人员，

## 8.生产管理

显示于左侧。

### 8.3 修改密码

点击“修改密码”会弹出如图 8-3 所示对话框，输入原密码和新密码即可修改密码。



修改密码对话框，包含以下输入项：

- 原密码: \_\_\_\_\_
- 新密码: \_\_\_\_\_
- 确认密码: \_\_\_\_\_

底部按钮：确认 | 取消

图 8-3 修改密码

### 8.4 程序管理

点击“程序管理”，会显示如图 8-4 所示界面，复选所需删除的程序文件即可删除所选定的程序文件。



图 8-4 程序管理界面

在图 8-4 所示界面，复选所需复制的程序文件，选择相应 U 盘或移动硬盘，即可复制所选定的程序文件至相应 U 盘或移动硬盘。

### 8.5 网络设置


点击“网络设置”，会弹出如图 8-5 所示界面，设置完成后，点击  图标即可返回原软件界面。



图 8-5 网络设置

## 8.6 报警历史

点击“报警历史”即可查看当前的报警历史列表。如图8-6所示：

2014-05-19 22:58:44组1( 3603):J4轴超速
2014-05-17 18:27:31组1( 3603):J4轴超速
2014-05-17 17:40:45组1( 3603):J4轴超速
2014-05-16 22:56:57组1( 3401):J2轴负向软限位
2014-05-16 00:59:12组1( 2000):急停
2014-05-16 00:33:45组1( 2000):急停
2014-05-16 00:30:02组1( 2000):急停
2014-05-16 00:26:05组1( 2000):急停
2014-05-14 00:58:13组1( 2000):急停
2014-05-12 19:17:11组1( 3302):J3轴正向软限位
2014-05-09 17:37:34组1( 3605):J6轴超速
2014-05-09 17:20:15组1( 3605):J6轴超速
2014-05-09 17:16:37组1( 3605):J6轴超速
2014-05-09 17:13:32组1( 3605):J6轴超速
2014-05-09 17:11:54组1( 3301):J2轴正向软限位
2014-05-09 17:09:19组1( 3301):J2轴正向软限位
2014-05-09 17:08:18组1( 3501):J2轴跟踪误差过大

关闭

图 8-6 报警历史显示

### 8.7 操作记录

点击“操作记录”即可查看当前的操作记录列表（包含自动运行、手动运行、修改参数等）。如图 8-7 所示：

2014-05-20 23:31:59自动运行:执行 启动
2014-05-20 23:31:49自动运行:执行 启动
2014-05-19 23:05:32自动运行:执行 启动
2014-05-19 23:03:03自动运行:执行 启动
2014-05-19 23:02:51自动运行:执行 启动
2014-05-19 23:00:29自动运行:执行 启动
2014-05-17 23:11:53自动运行:执行 启动
2014-05-16 22:58:59修改参数 + [code:60000[value: w]
2014-05-16 22:57:57修改参数 + [code:61500[value: 6]
2014-05-16 18:41:07修改参数 + [code:61542[value: 0]
2014-05-16 01:12:36自动运行:执行 启动
2014-05-13 17:46:45自动运行:选择 单周
2014-05-13 17:46:44自动运行:执行 停止
2014-05-13 17:46:27自动运行:执行 启动
2014-05-13 17:45:59自动运行:执行 启动
2014-05-13 17:40:10自动运行:执行 启动
2014-05-13 02:41:07自动运行:执行 启动

关闭

图 8-7 操作记录显示

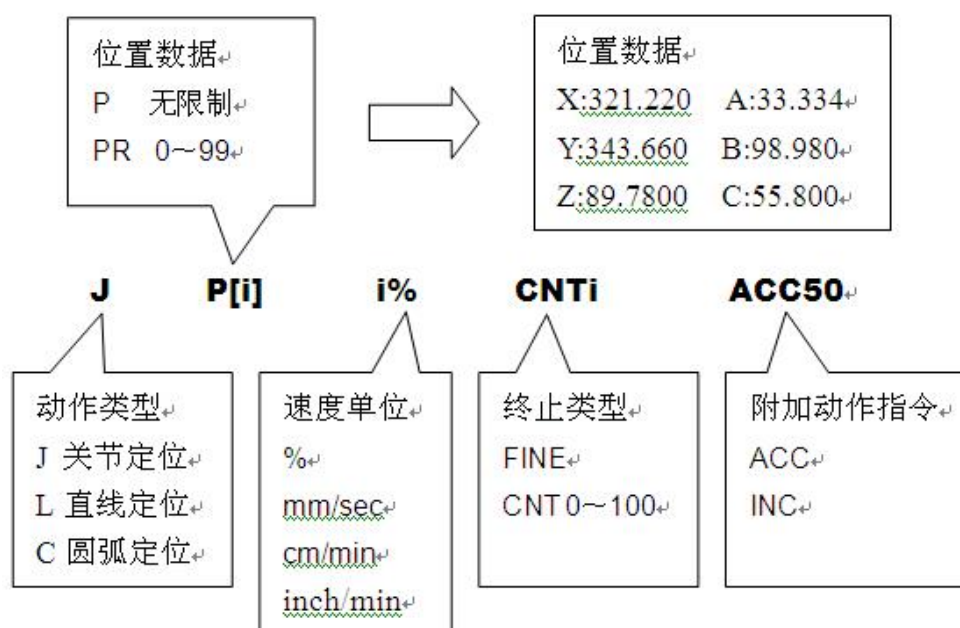
## 第二部分 编程篇

## 9 运动指令

运动指令实现以指定速度、特定路线模式等将工具从一个位置移动到另一个指定位置。在使用运动指令时需指定以下几项内容：

- (1) 动作类型：指定采用什么运动方式来控制到达指定位置的运动路径；
- (2) 位置数据：指定运动的目标位置；
- (3) 进给速度：指定机器人运动的进给速度；
- (4) 定位路径：指定相邻轨迹的过渡形式；
- (5) 附加运动指令：指定机器人在运动过程中的附加执行指令。

运动指令格式如下所示：



在程序示教的过程中，使用菜单树中的“运动指令”即可添加标准的运动指令，详细操作步骤请参阅前面章节的操作说明。

### 9.1 动作类型

动作类型，即指定到达指定位置的运动路径，机器人运动的类型有三种：关节定位 (J)、直线运动 (L)、圆弧运动 (C)。

#### 1) 关节定位

关节定位是移动机器人各关节到达指定位置的基本动作模式。独立控制各个关节同时运动到目标位置，即机器人以指定进给速度，沿着（或围绕）所属轴的方向，同时加速、减速或停止。工具的运动路径通常是非线性的，在两个指定的点之间任意运动。以最大进给速度的百分数作为关节定位的进给速度，其最大速度由参数设定，程序指令中只给出实际运动的倍率。关节定位过程中没有控制被驱动的工具的姿态。



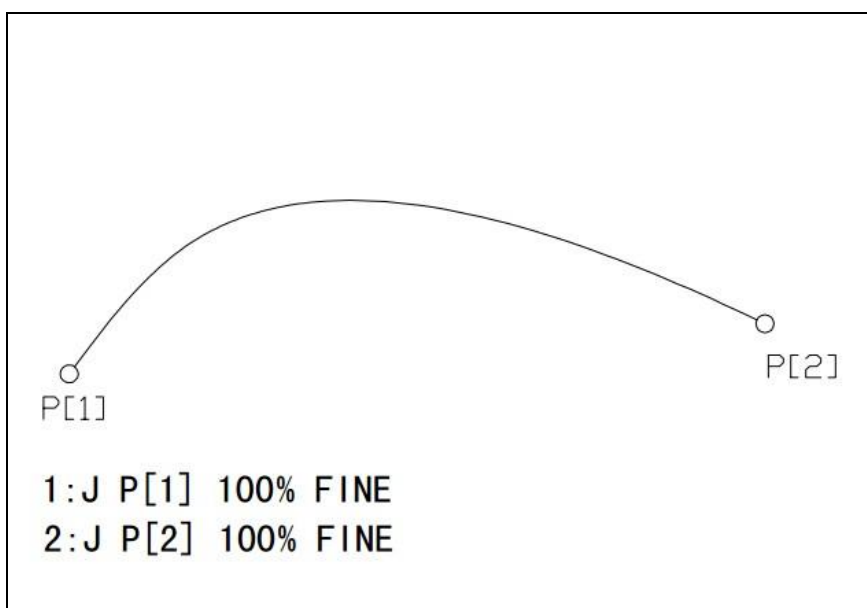


图 9-1 J 关节定位

图中由 P[1] 点以 100%速度采用关节定位方式移动至 P[2] 点。

## 2) 直线运动 L

直线运动指令控制 TCP（工具中心点）沿直线轨迹运动到目标位置，其速度由程序指令直接指定，单位可为 mm/sec、cm/min、inch/min。通过区别起点和终点时的姿态，来控制被驱动的工具的姿态。

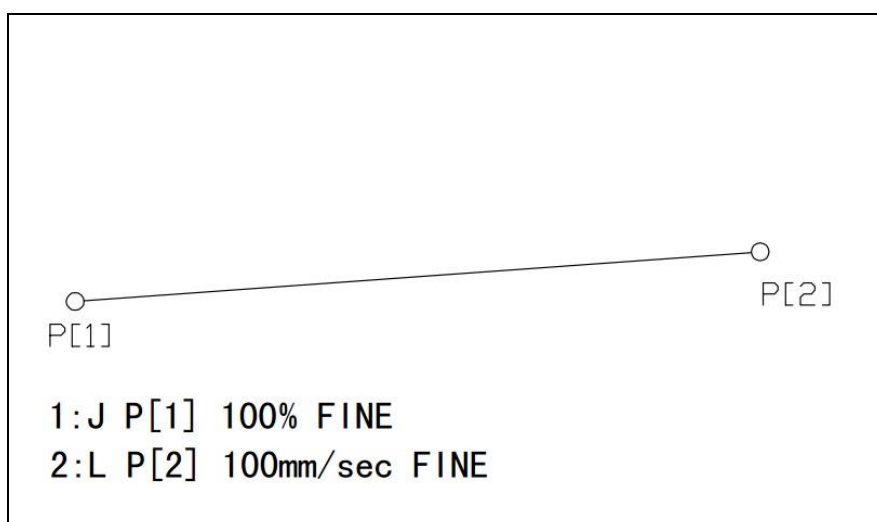


图 9-2 L 直线运动

图中由 P[1] 点以 100mm/sec 的速度采用直线运动方式移动至 P[2] 点。

## 3) 圆弧运动 C

圆弧运动指令控制 TCP（工具中心点）沿圆弧轨迹从起始点经过中间点移动到目标位置，中间点和目标点在指令中一并给出。其速度由程序指令直接指定，单位可为 mm/sec、cm/min、inch/min。通过区别起点和终点时的姿态，来控制被驱动的工具的姿态。

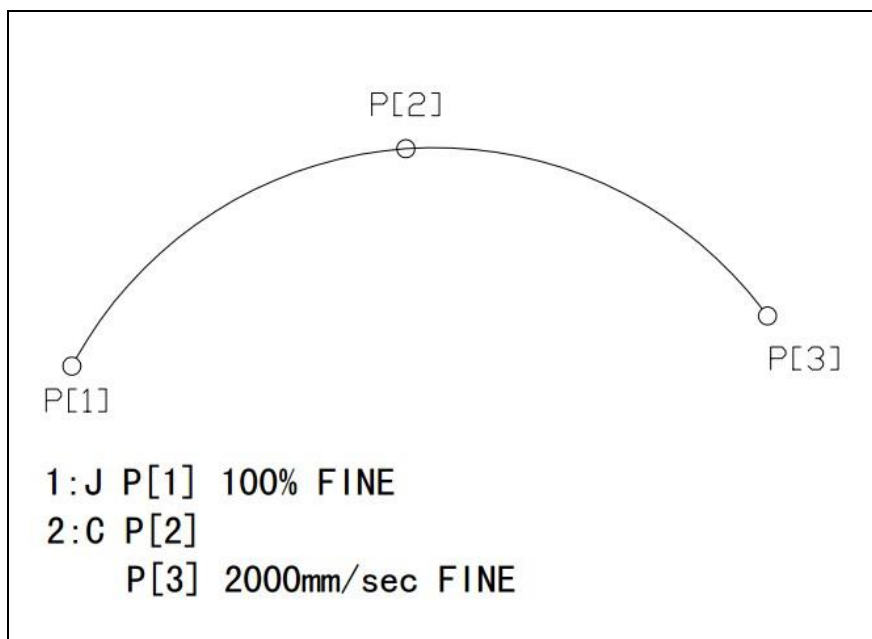


图 9-3 C 圆弧定位

图中由 P[1]点开始沿着过 P[2]点的圆弧以 2000mm/sec 的速度运动至 P[3]点

## 9.2 位置数据

位置数据包括位置和机器人的姿态。在进行运动指令示教时，位置数据也同时被写入程序文件。

位置数据被划分为两种类型。一种是在关节坐标系下的关节坐标（J1、J2、J3、J4、J5、J6），没有姿态信息；另一种类型在直角坐标系下的位置和姿态两种信息（X、Y、Z、A、B、C）。

### 1) 直角坐标系

直角坐标系下的位置数据包含四个元素：用户坐标系序号、工具坐标系序号、位置/姿态和配置。

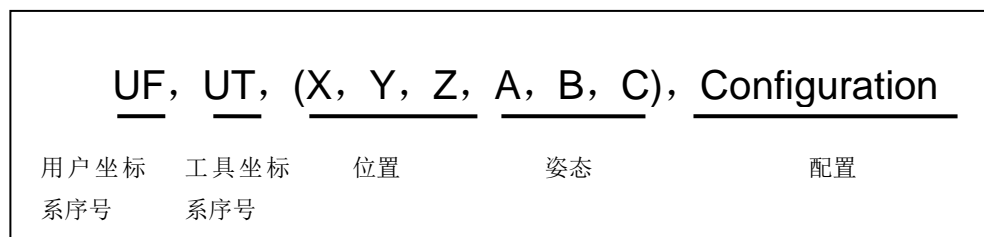


图 9-4 位置数据（直角坐标）

位置（X, Y, Z）：代表在直角坐标系下工具中心（TCP，即工具坐标系的原点）的三维坐标；

姿态（A, B, C）：代表在直角坐标系下，绕 X 轴、Y 轴、Z 轴的角位移。

配置：指机器人的姿态。包括关节定位和转数两种类型。

关节定位的配置：机械腕俯仰、机械臂的上面或下面、机械臂的前面或后面。

对于关节型六轴机器人，一般有三个奇异形位。以这三个奇异行为作为临界点，可以定义机器人的三个关节属性  $cfg1$ 、 $cfg2$ 、 $cfg3$ 。

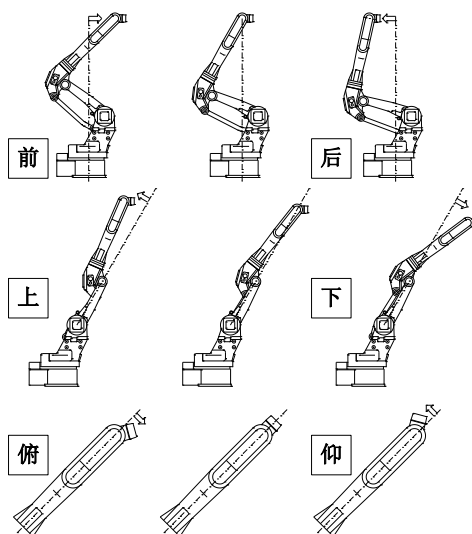


图 9-5

如图9-5图所示，以机器人第一关节的转动轴线与第二关节的轴线平行线两者所构成的平面为参考平面，当机器人手腕中心点位于平面两侧时，分别定义机器人的第一个关节属性为“前”( $cfg1=0$ )和“后”( $cfg1=1$ )；以机器人大臂和第三关节的轴线所构成的平面作为参考平面，当手腕中心点处于该参考平面两边时分别定义机器人的第二个关节属性为“上”( $cfg2=0$ )和“下”( $cfg2=1$ )；以机器人小臂和第五关节的轴向所构成的平面作为参考平面，当手腕中心点处于该参考平面两边时分别定义机器人的第二个关节属性为“俯”( $cfg3=0$ )和“仰”( $cfg3=1$ )。这三个关节属性恰好可以组合为“前下俯”、“前下仰”、“前上俯”、“前上仰”、“后上俯”、“后上仰”、“后下俯”、“后下仰”8种关节属性的组合。本控制系统通过关节属性来确定机器人在直角坐标系下的姿态。

## 2) 关节坐标

关节坐标系下的位置数据用每个关节的角度位置定义，关节坐标系位于每个关节的基准面。

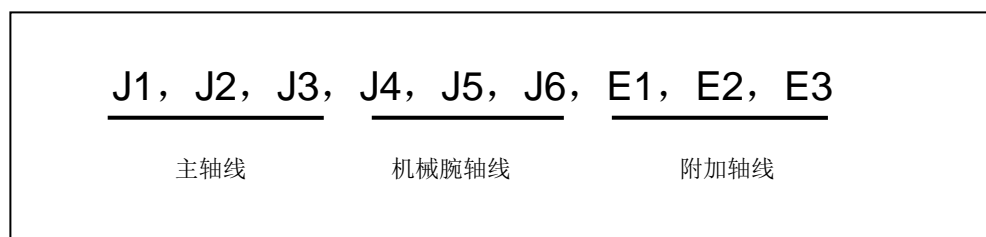


图 9-6 位置数据（关节坐标）

## 3) 位置变量和位置寄存器

在运动指令中，位置数据通过位置变量 ( $P[i]$ ) 和位置寄存器 ( $PR[i]$ ) 表示。一般

## 9.运动指令

情况下，使用位置变量。

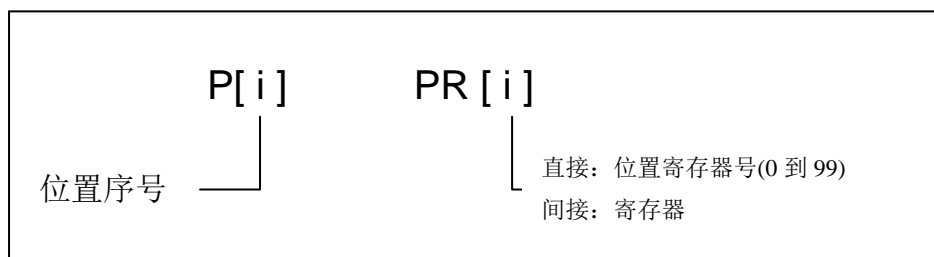


图 9-7 位置变量和位置寄存器

- (1) 位置变量：是指用于保存位置数据的变量。在示教过程中，位置数据被自动保存到程序文件中，此时的坐标系均为当前所选择的，当复制指令时，位置及相关信息也一同被复制。位置变量的取值范围无限制。
- (2) 位置寄存器：用于存放位置数据，类似通用寄存器（R 寄存器）一样使用。在菜单树窗口的“寄存器”下的“位置寄存器”中，可以对位置寄存器组号、属性等进行查看、设置和修改。

示例：

```
1:J P[12] 30% FINE
2:L PR[1] 300mm/sec CNT50
3:L PR[R[3]] 300mm/sec CNT50
```

### 9.3 进给速度

进给速度指机器人的运动速度。在程序执行过程中，进给速度可以通过倍率进行修调。倍率值范围为0%到150%。进给速度单位取决于动作指令类型。

**J P[1] 50% FINE //机器人以最大进给速度的 50%采用关节定位方式移动至 P[1]**  
当动作类型为关节定位时，指定最大进给速度的百分数，范围从1%到100%；

**L P[1] 100mm/sec FINE//机器人以 100 毫米每秒的速度采用直线运动方式移动至 //P[1]**

如果指定的动作类型为直线运动或者圆弧运动时，直接指定运动的速度值 (mm/sec、cm/min、inch/min)，最大值由参数限制；

**用寄存器指定进给速度**

进给速度可以用寄存器指定。允许用户在寄存器中计算好进给速度后，再为动作指令指定进给速度。由于此时的进给速度取决于指定的寄存器，这就意味着机器人可能以一种出乎意料的速度运动，所以在使用这个功能时，在示教和操作过程中都必须小心谨慎的指定寄存器。

示例:

```
1:J P[1] R[1]% FINE
2:L P [2] R[2]mm/sec FINE
3:C P [1]
    P[2] R[3]mm/sec CNT50
```

## 9.4 定位路径

定位路径指定了相邻轨迹间的过渡形式，有以下两种形式：

- (1) FINE：相当于准确停止。
- (2) CNT：相当于圆弧过渡，CNT 后的数值为过渡误差，该数值的取值范围为 0到100。CNT0等价于 FINE。

**FINE 定位路径： J P[i] 50% FINE**

当指定 FINE 定位路径时，机器人在向下一个目标点驱动前，停止在当前目标点上。

**CNT 定位路径： J P[i] 50% CNT50**

当指定 CNT 定位路径时，机器人逼近一个目标点但是不停留在这个目标点上，而是向下一个目标点移动。其取值为逼近误差。例如 CNT50，表示目标 P[i]点到机器人实际运行路径的最短距离为50mm。

**提示：**

- (1) 示教如等待指令这样的指令时，机器人应停止在目标点上来执行该指令，即使用 FINE 定位路径。
- (2) 一些短距离、高速度的动作，使用 CNT 定位路径连续不断的执行可以减速，即使指定的 CNT 值很大时，也可以减速。

使用 CNT 和 FINE 时，机器人运动路径如图9-8所示：

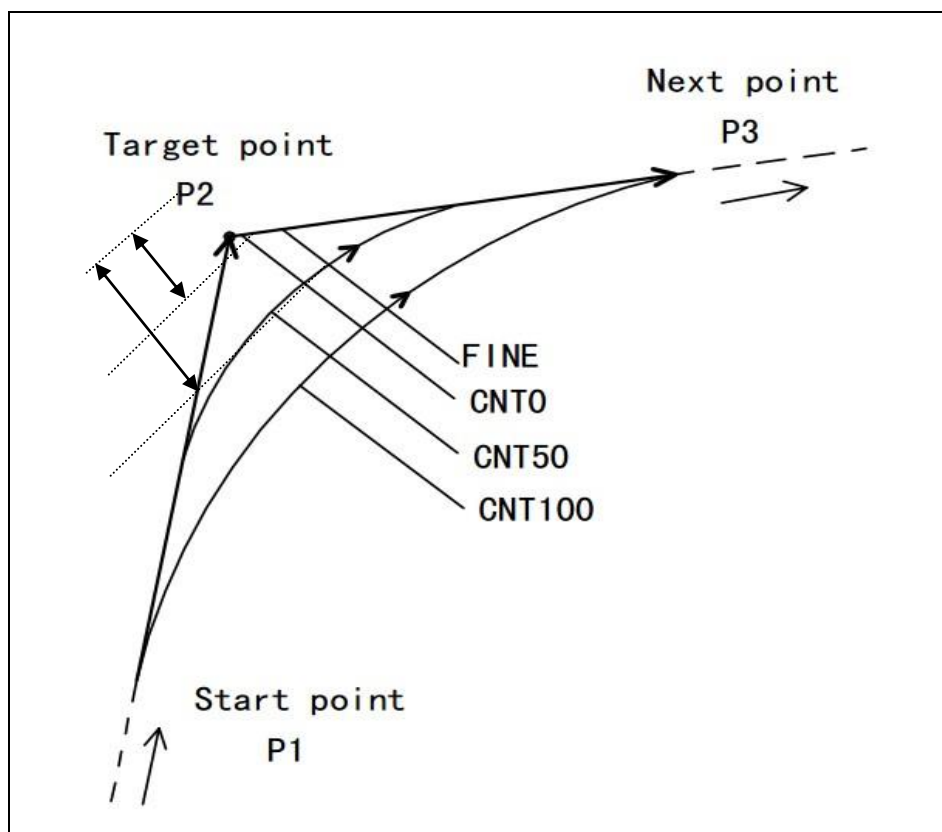


图 9-8

### 9.5 附加动作指令

附加动作指令让机器人完成特殊的任务，本系统目前支持的附加指令有：

- 加速倍率（ACC）
- 增量指令（INC）

#### 1) 加速倍率（ACC）

该指令指定运动过程中的加速度的倍率。ACC 后紧跟数字，表示加速度的倍率。如 ACC80，即80%的加速度。当减小加速倍率时，加速的时间会变长（加速和减速慢慢的完成）。当加速倍率提高时，加速时间就会变短（加速和减速快速的完成）。

从起点到目标点，用于执行动作的时间取决于加速倍率。加速倍率值的范围在 1% 到 2000%之间变化。在目标位置处添加加速倍率。

加速倍率值对比如图9-9所示：

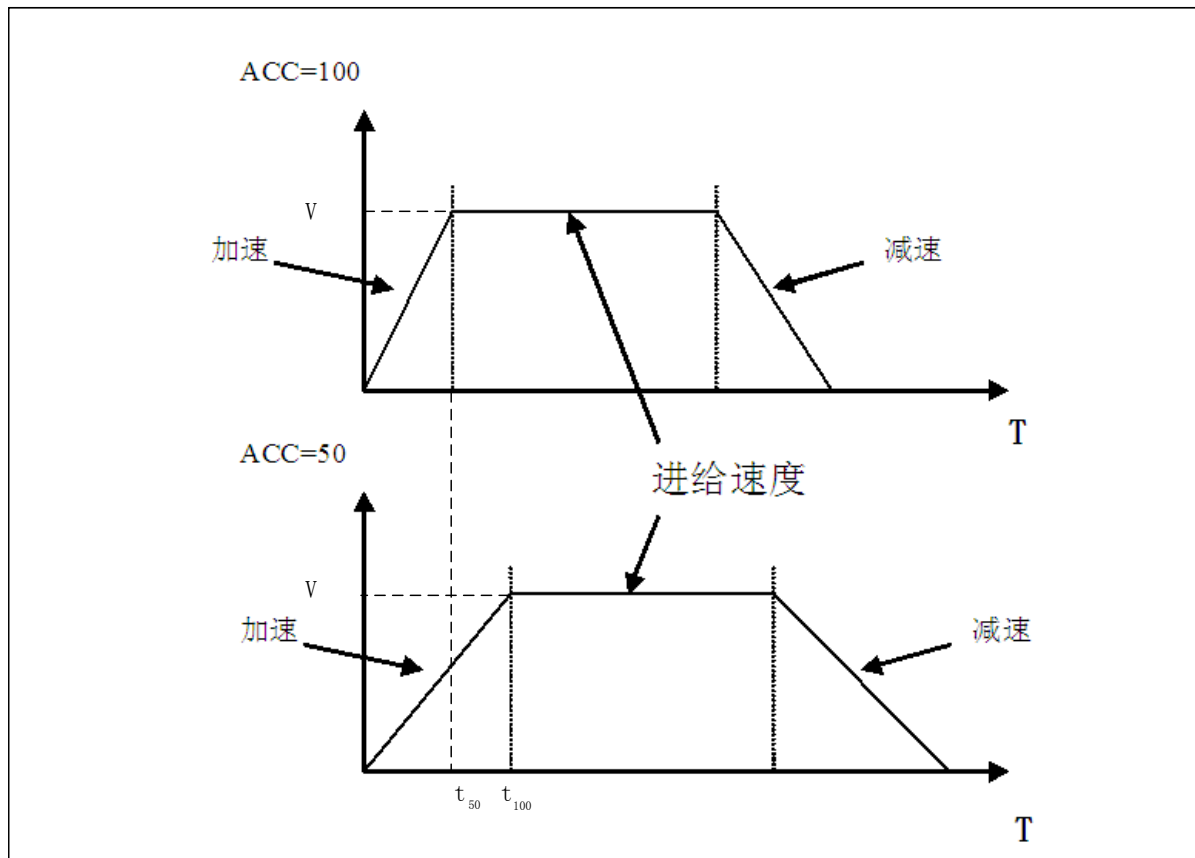


图 9-9 加速倍率值对比

图中 ACC 越大，加速达到进给速度所用时间越短

**注：**如果加速倍率很大，可能会发生剧烈的颤动，从而引起伺服报警。如果是因为增加了加速倍率指令而导致上述情况的发生，要么减小加速倍率值，要么删除加速倍率指令。

## 2、增量指令（INC）

**示例：**LP[1] 500mm/sec FINE INC

增量指令将运动指令中的位置数据用作当前位置的增量，即增量指令中的位置数据为机器人移动的增量。

- 当位置数据为关节坐标值时，提供了每个轴的增量数据。
- 当位置变量（P[i]）作为位置数据时，用户坐标系的基准通过用户坐标系的序号指定，而用户坐标系的序号是在位置数据中指定的。
- 当位置寄存器作为位置数据时，基准坐标系即为当前用户坐标系。

# 10 寄存器指令

寄存器指令在寄存器上完成算术运算。本机器人控制系统支持两种寄存器指令，分别为：R 寄存器指令和位置寄存器指令。

在寄存器操作中，支持以下多项式运算：

示例：1:  $R[2]=R[3]-R[4]+R[5]-R[6]$

2:  $R[10]=R[2]*100/R[6]$

寄存器指令支持多项式运算，但使用时请注意：单独一行最多5个运算符。

本系统支持+、-、\*、/、MOD、DIV 在一行中混合使用。在多项式中，MOD、DIV 属于同一级别，\*、/属于同一级别，+、-属于同一级别，优先级别划分为：

- MOD、DIV优先级高于\*、/；
- \*、/优先级高于+、-；
- 同一级别下，优先级从左到右，即左边高于右边；

## 10.1 寄存器（R）指令

### 说 明

寄存器指令在寄存器上完成算术运算。寄存器是一个存储数据的变量，本机器人控制系统提供 200 个 R 寄存器。

### 指令格式

#### 1) $R[i]=(value)$

$R[i]=(value)$ 指令把数值(value)赋值给指定的 R 寄存器。

其中，i 的范围是0到199，(value)可以取常数(constant)、寄存器(R)、位置寄存器中的某个轴(PR[i,j])、数字量输入/输出(DI[i]/DO[i])、模拟量输入/输出(AI[i]/AO[i])。

示例：

1:  $R[1] = DI[3]$

2:  $R[R[4]] = AI[R[1]]$

#### 2) $R[i]=(value)+(value)$

$R[i]=(value)+(value)$ 指令把两个数值的和赋值给指定的 R 寄存器。



**3)  $R[i]=(value)-(value)$** 

$R[i]=(value)-(value)$ 指令把两个数值的差赋值给指定的 R 寄存器。

**4)  $R[i]=(value)*(value)$** 

$R[i]=(value)*(value)$ 指令把两个数值的乘积赋值给指定的 R 寄存器。

**5)  $R[i]=(value)/(value)$** 

$R[i]=(value)/(value)$ 指令把两个数值的商赋值给指定的 R 寄存器。

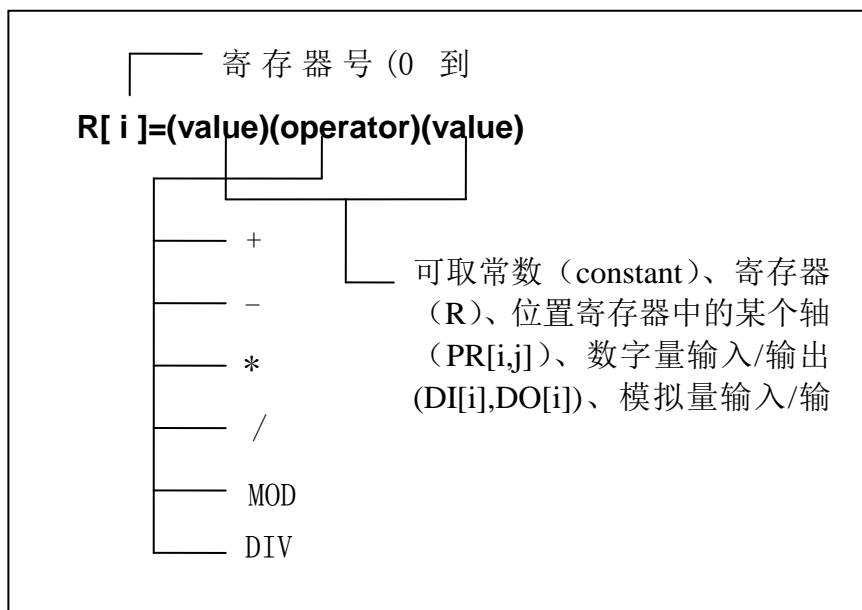
**6)  $R[i]=(value)MOD(value)$** 

$R[i]=(value)MOD(value)$ 指令把两个数值的商的余数（小数部分）赋值给指定的 R 寄存器。

**7)  $R[i]=(value)DIV(value)$** 

$R[i]=(value)DIV(value)$ 指令把两个数值的商（整数部分）赋值给指定的 R 寄存器。

对于运算寄存器指令，可归纳如下：



示例：

1:  $R[3] = DI[4] + PR[1,2]$

2:  $R[R[4]] = R[1] + 1$

## 10.2 位置寄存器 (PR) 指令

### 说 明

位置寄存器指令在位置寄存器上完成算术操作。位置寄存器指令可以把位置数据、两个数值的和、差赋值给指定的位置寄存器。

## 10.寄存器指令

位置寄存器是一个存储位置数据(x、y、z、w、p、r)的变量，本系统提供100个位置寄存器。

### 指令格式

#### 1) PR[i]=(value)

PR[i]=(value)指令把数值(value)赋值给指定的位置寄存器。

其中，i 的范围是0到99，(value)可以取位置寄存器 (PR)、位置变量 (P)、直角坐标系中的当前位置 (Lpos)、关节坐标系中的当前位置 (Jpos)、用户坐标系 (UFRAME[i])、工具坐标系 (UTOOL[i])。

示例：

- 1: PR[1] = Lpos
- 2: PR[R[4]] = UFRAME[R[1]]
- 3: PR[9] = UTOOL[1]

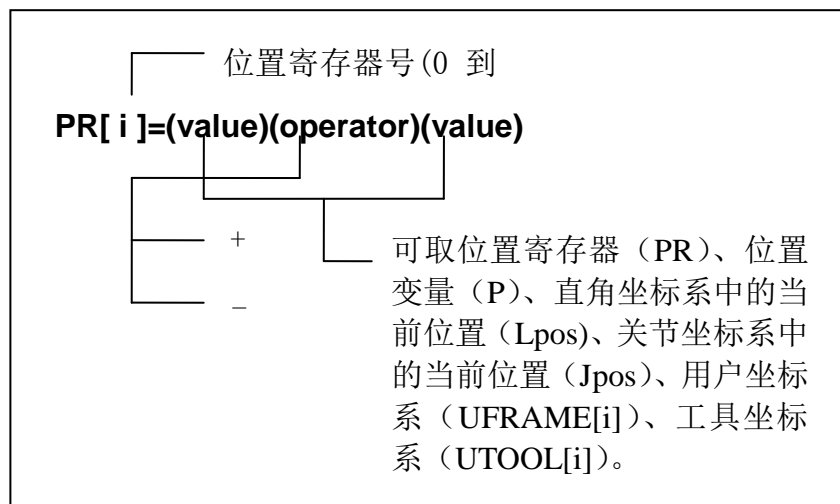
#### 2) PR[i]=(value)+(value)

PR[i]=(value)+(value) 指令把两个数值的和赋值给指定的位置寄存器。

#### 3) PR[i]=(value)-(value)

PR[i]=(value)-(value) 指令把两个数值的差赋值给指定的位置寄存器。

对于运算位置寄存器指令，可归纳如下：



示例：

- 1: PR[3] = PR[4]+Lpos
- 2: PR[4] = PR[R[1]]

## 10.3 位置寄存器轴指令

### 说 明

位置寄存器轴指令在位置寄存器上完成计算操作。PR[i, j]中的元素 i 代表位置寄存器的序号, j 代表位置寄存器元素序号。位置寄存器轴指令可以将位置数据元素的值, 或两个数据的和、差、商、余数等赋值给指定的位置寄存器元素。

PR[i, j]类型如下:

<b>PR[ i, j ]</b>	
位置寄存器 号	<div style="display: flex; align-items: center;"> <div style="border-left: 1px solid black; border-bottom: 1px solid black; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border-left: 1px solid black; border-bottom: 1px solid black; width: 20px; height: 20px; margin-right: 5px;"></div> </div> 直角坐标系下, 取值为: 0=X, 1=Y, 2=Z, 3=A, 4=B, 5=C; 关节坐标系下, 取值为: 0=J1, 1=J2, 2=J3, 3=J4, 4=J5, 5

### 指令格式

#### 1) PR[i, j]=(value)

PR[i, j]=(value)指令把数值(value)赋值给指定的位置寄存器元素。

其中, i 的范围是0到99, (**value**)可以取常数 (**constant**)、寄存器 (**R**)、位置寄存器中的某个轴 (PR[i,j])、位置变量中的某个轴 (P[i,j])、数字量输入/输出(DI[i]/DO[i])、模拟量输入/输出(AI[i]/AO[i])。

示例:

1: PR[1, 2] = R[3]

2: PR[4, 3] = 324.5

#### 2) PR[i,j]=(value)+(value)

PR[i,j]=(value)+(value)指令把两个数值的和赋值给指定的位置寄存器元素。

#### 3) PR[i,j]=(value)-(value)

PR[i,j]=(value)-(value)指令把两个数值的差赋值给指定的位置寄存器元素。

#### 4) PR[i,j]=(value)\*(value)

PR[i,j]=(value)\*(value)指令把两个数值的乘积赋值给指定的位置寄存器元素。

#### 5) PR[i,j]=(value)/(value)

PR[i,j]=(value)/(value)指令把两个数值的商赋值给指定的 R 寄存器位置寄存器元素。

## 10.寄存器指令

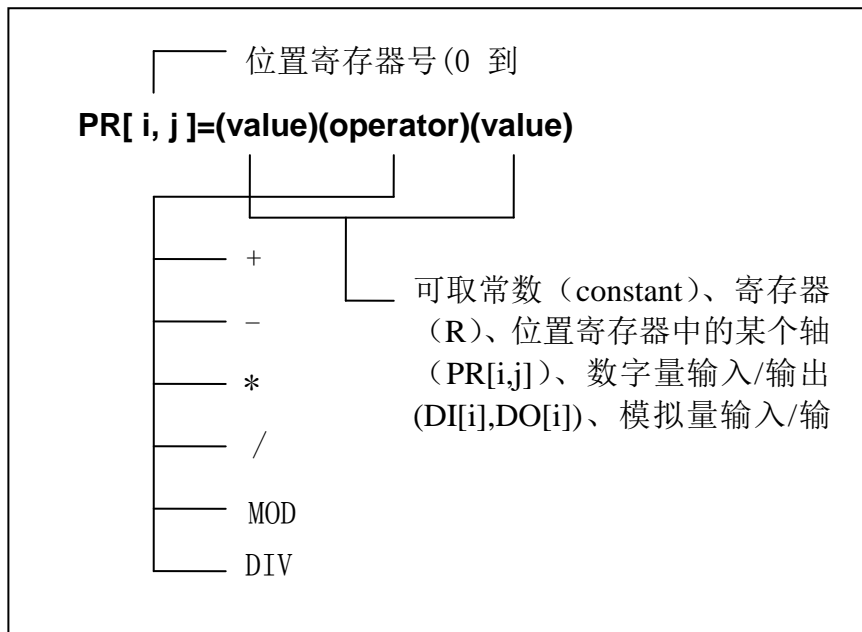
### 6) $PR[i,j]=(value)MOD(value)$

$PR[i,j]=(value)MOD(value)$ 指令把两个数值的商的余数赋值给指定的位置寄存器元素。

### 7) $PR[i,j]=(value)DIV(value)$

$PR[i,j]=(value)DIV(value)$ 指令把两个数值的商的整数赋值给指定的位置寄存器元素。

对于运算位置寄存器轴指令，可归纳如下：



示例：

1:  $PR[3, 5] = R[3]+DI[4]$

2:  $PR[4, 3] = PR[1, 3] - 3.528$

## 11 I/O 指令

输入输出指令用于操作 IO 的状态（读取输入或设置输出）。

- 数字信号输入/输出指令
- 模拟信号输入/输出指令

### 11.1 数字输入/输出（DI/DO）指令

#### 说 明

数字输入指令（DI）和数字输出指令（DO）是可以被用户控制的输入输出信号。

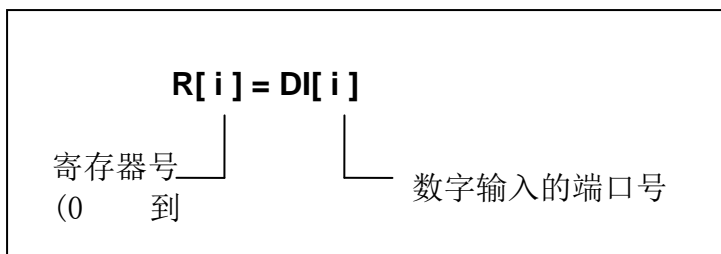
#### 指令格式

##### 1) 读操作

###### (1) $R[i]=DI[i]$

$R[i]=DI[i]$ 指令把数字输入信号（ON=1 / OFF=0）赋值给指定的 R 寄存器。

指令结构如下：



示例：

1:  $R[1]=DI[1]$

2:  $R[R[3]]=DI[R[4]]$

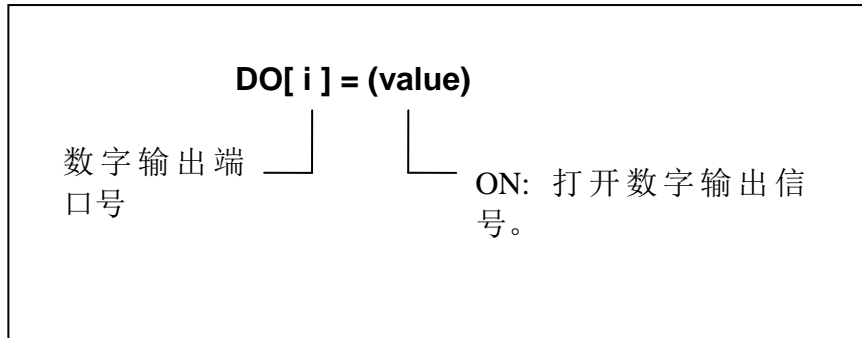
##### 2) 写操作

###### (1) $DO[i]=ON/OFF$

$DO[i]=ON/OFF$  指令把 ON=1 / OFF=0赋值给指定的数字输出信号。

指令结构如下：

## 11.I/O 寄存器指令

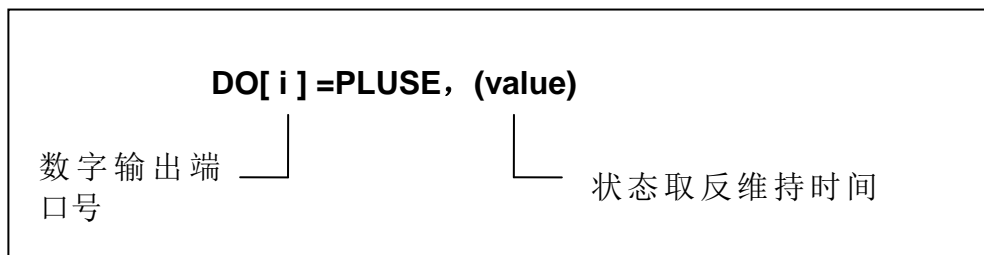


示例:

- 1: DO[1] = ON
- 2: DO[ R[3] ] = OFF

### (2) DO[i]=PLUSE, (value)

DO[i]=PLUSE, (value)指令使 DO[i]的状态取反, 并维持一段指定时间 value。  
指令结构如下:



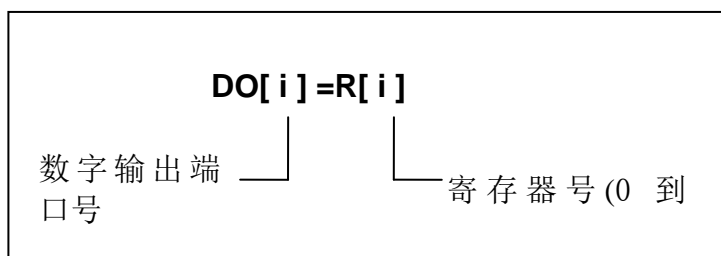
示例:

- 1: DO[2] = PULSE, 0.2sec
- 2: DO[ R[3] ] = PULSE, 1.2sec

### (3) DO[i]=R[i]

DO[i]=R[i]指令根据指定寄存器 R 的值, 设置指定的数字输状态。当指定的 R 寄存器的值为0时, 数字输出 OFF; 当为非0, 数字输出 ON。

指令结构如下:



示例:

- 1: DO[1] = R[2]

2: DO[ R[5] ] = R[R[1]]

## 11.2 模拟输入/输出 (AI/AO) 指令

### 说 明

模拟量输入输出值为连续值，信号的幅值可代表温度、电压或其他数据。

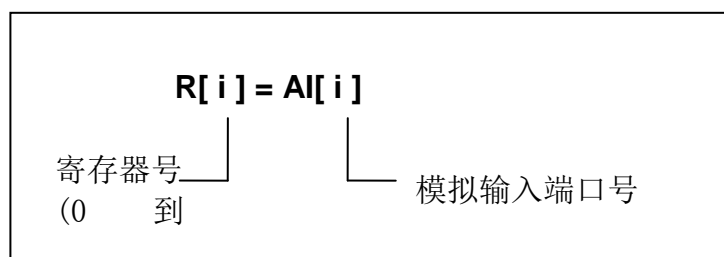
### 指令格式

#### 1、读操作

##### (1) $R[i]=AI[i]$

$R[i]=AI[i]$ 指令将模拟输入信号赋值给指定的 R 寄存器。

指令结构如下：



示例：

1:  $R[1] = AI[1]$

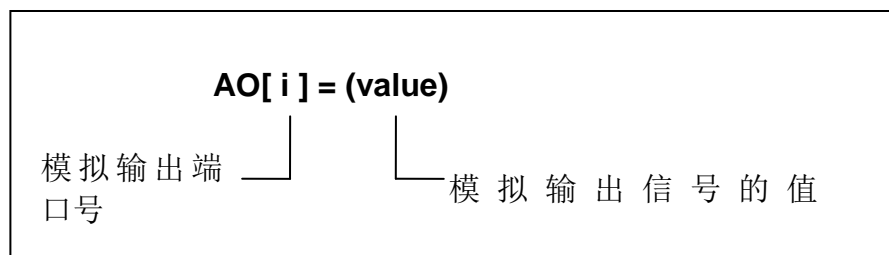
2:  $R[ R[3] ] = AI[ R[4] ]$

#### 2、写操作

##### (1) $AO[i]=(value)$

$AO[i]=(value)$ 指令将数值(value)作为指定的模拟输出信号的值。

指令结构如下：



示例：

1:  $AO[1] = 0$

## 11.I/O 寄存器指令

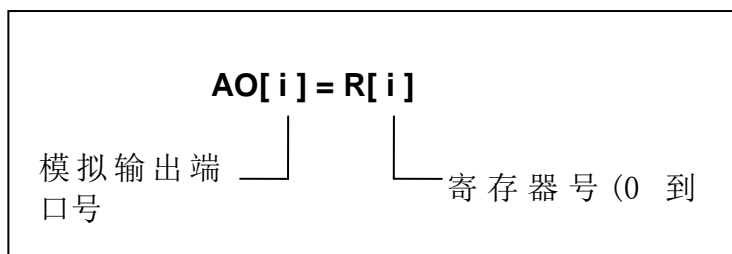
---

2: AO[ R[3] ] = 3276

(2) **AO[i]=R[i]**

AO[i]=(value)指令将数值(value)作为指定的模拟输出信号的值。

指令结构如下:



示例:

1: AO[1] = R[2]

2: AO[ R[5] ] = R[R[1]]





## 12 条件指令

条件指令即条件比较指令，当某些条件满足时，在指定的标签或者程序里产生分支。条件比较指令包括寄存器条件比较指令和输入输出条件比较指令。

比较运算符包括：

- >: 大于
- >=: 大于或等于
- =: 等于
- <=: 小于或等于
- <: 小于
- <>: 不等于

对于寄存器（R）、模拟量输入/输出比较指令，可使用全部的比较符：>、>=、=、<=、<、<>；但对于数字量输入/输出比较时，只能使用=（等于）和<>（不等于）两种比较符。

### 12.1 寄存器条件比较指令

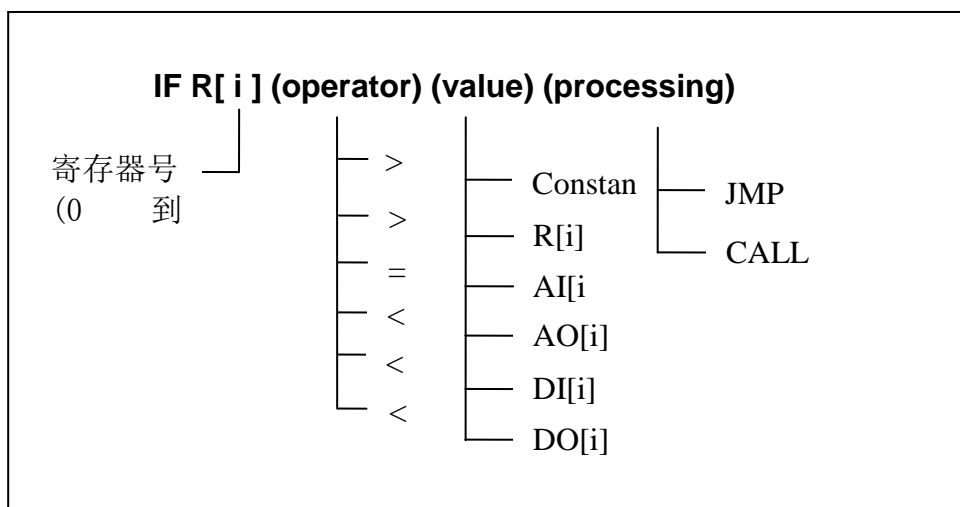
#### 说 明

寄存器条件比较指令将存储在寄存器中的值与另一个值比较。当比较条件满足时，执行指定的操作。

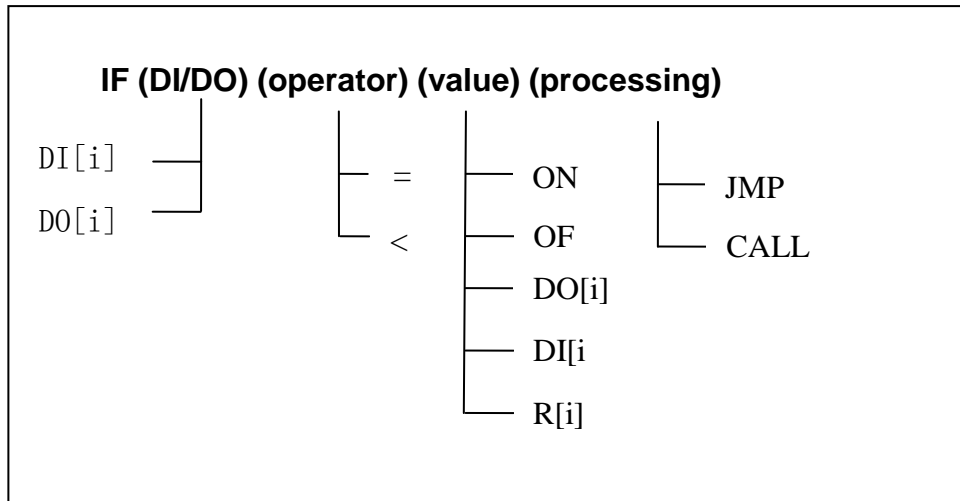
#### 指令格式

**IF R[i] (运算符) (value) (操作)**

指令结构如下：







示例：

- 1: IF DO[2] <> OFF, JMP LBL[1]
- 2: IF DI[R[3]] = R[R[4]], CALL subprog2

### 12.3 复合条件的使用

在比较条件中，可以使用逻辑与（AND）和逻辑或（OR）来指定复合条件。这样就简化了程序的结构，使得条件的比较更加高效。

- 逻辑与(AND)

IF <条件1> and <条件2> and <条件3>, JMP LBL [3]

- 逻辑或(OR)

IF <条件1> or <条件2>, JMP LBL [3]

如果逻辑与（AND）和逻辑或（OR）同时出现在一个指令中，逻辑就会变复杂，从而削弱了程序的可读性和可编辑性。因此，禁止在一个条件指令中同时使用 AND 和 OR。

如果在单行上为一个指令指定多个“AND”或“OR”逻辑运算，并且这些逻辑运算中有一个是从“AND”变成“OR”（或是从“OR”变成“AND”），那么其他所有的“AND”（或者“OR”也会相应的改变）。

每行中最多允许使用5个“AND”（或者“OR”）逻辑运算符。

示例： 1: IF R[1]>234 AND DO[R[2]]<>OFF AND AI[3]>=R[R[4]], JMP LBL[1]

2: IF DO[2]<>OFF OR AO[R[3]]>=R[4], CALL subprog1

## 13 等待指令

等待指令用于在一个指定的时间段内，或者直到某个条件的满足时的时间段内，结束程序的指令，等待指令包括如下两种：

- 指定时间的等待指令：等待一个指定的时间后，再执行后续程序；
- 条件等待指令：等待指定的条件满足后，再执行后续程序。

### 13.1 指定时间的等待指令

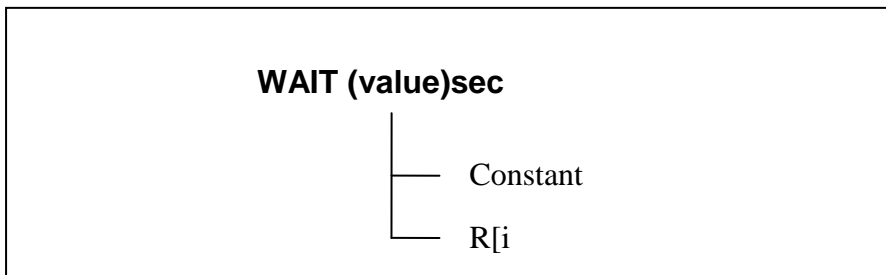
#### 说 明

指定时间的等待指令，等待一个指定的时间（以秒为单位）后，再执行后续程序。

#### 指令格式

##### **WAIT (value)sec**

指令结构如下：



示例：

- 1: WAIT 10.5sec
- 2: WAIT R[1]sec

### 13.2 条件等待指令

条件等待指令，等待程序的执行，直到指定的条件满足或者经过了指定的时间段。有以下两种指定暂停程序的方法：

- 如果没有指定操作（processing），程序将无限期等待，直到满足指定条件为止；
- 如果指定了操作（如TIMEOUT LBL[i]），当指定的条件不满足，且等待超时后，程序将跳转到指定的目标处运行。超时等待时间由系统参数设置。

## 13.等待指令

### 13.2.1寄存器条件等待指令

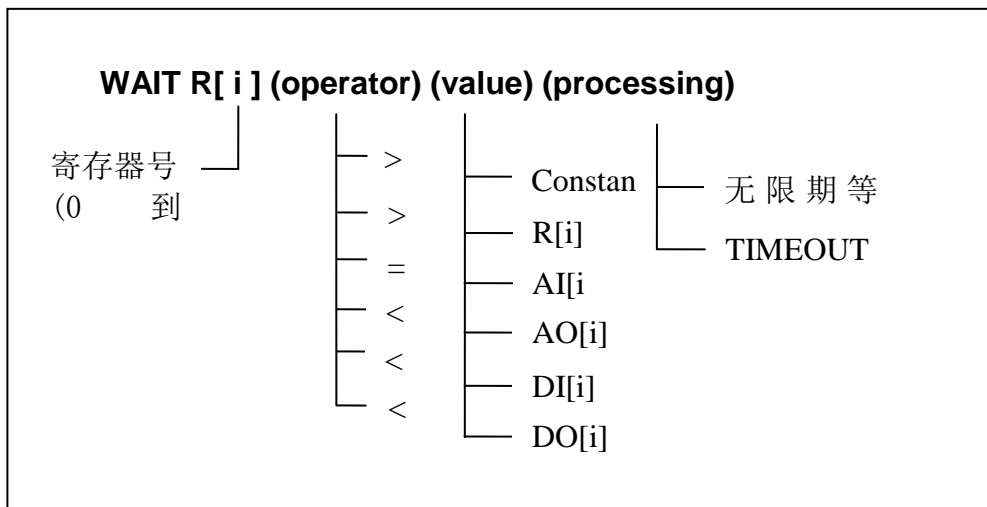
#### 说 明

寄存器条件等待指令将 R 寄存器的值与另外一个值进行比较，并等待直到满足比较条件为止。

#### 指令格式

**WAIT R[i] (比较符) (value) (操作)**

指令结构如下：



示例：

1: WAIT R[2] <> 1, TIMEOUT LBL[1]

2: WAIT R[R[1]] >= R[3]

### 13.2.2输入/输出条件等待指令

#### 说 明

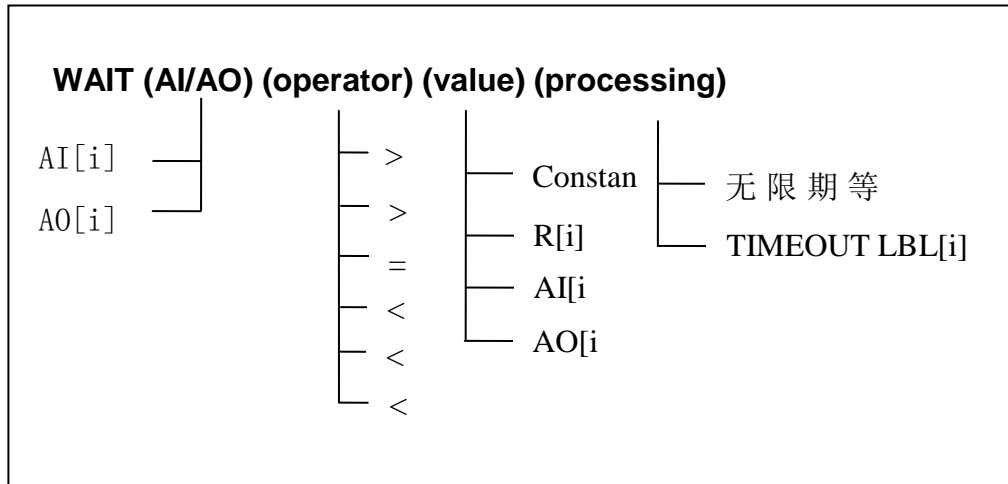
输入/输出条件等待指令将输入/输出信号的值与另一个值进行比较，并等待直到满足比较条件为止。

#### 指令格式

(1) 模拟输入输出条件等待指令

**WAIT (AI/AO) (比较符) (value) (操作)**

指令结构如下：



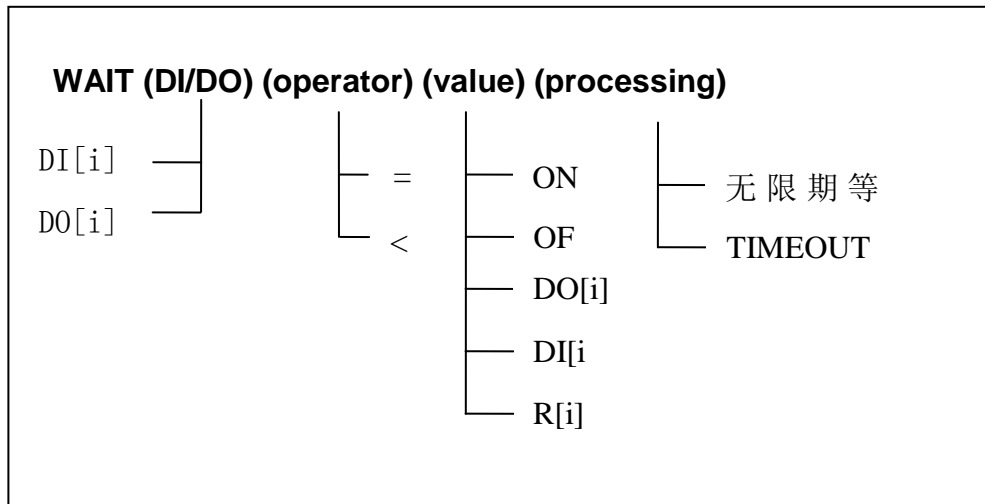
示例:

- 1: WAIT AI[2] <> 1, TIMEOUT LBL[1]
- 2: WAIT AO[R[1]] >= R[3]

## (2) 数字输入输出条件比较指令

### WAIT (DI/DO) (比较符) (value) (操作)

指令结构如下:



示例:

- 1: WAIT DI[2] <> ON, TIMEOUT LBL[1]
- 2: WAIT DO[R[1]] = R[3]

# 14 流程控制指令

流程控制指令用来控制程序的执行顺序，控制程序从当前行跳转到指定行去执行，流程控制指令包括以下几种指令：

- 标签指令；
- 程序结束指令；
- 无条件跳转指令；
- 子程序调用指令。

## 14.1 标签指令

### 说 明

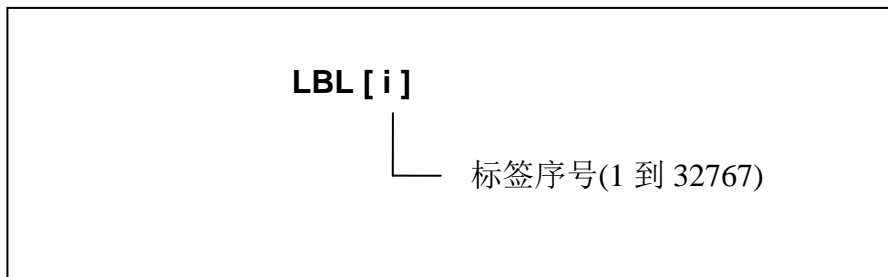
标签指令用于指定程序执行的分支跳转的目标。

标签一经执行，对于条件指令、等待指令和无条件跳转指令都是适用的。不能把标签序号指定为间接寻址（如 LBL[R[1]]）。

### 指令格式

#### LBL[i]

指令结构如下：



示例：

1: LBL[2]

## 14.2 程序结束指令

### 说 明

程序结束指令标志着一个程序的结束。通过这个指令终止程序的执行。如果该程序是被其他的主程序调用，则控制该子程序返回到主程序中。

程序结束指令在新建程序时，系统已自动添加到程序文件的末尾，无需用户自己添加。



## 指令格式

**END**

示例：

1:END

## 14.3 无条件跳转指令

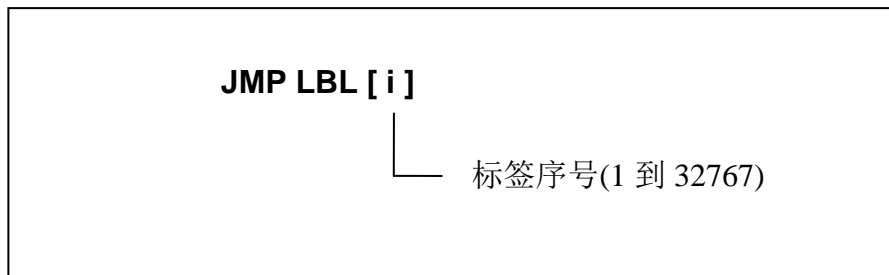
### 说 明

无条件跳转指令是指在同一个程序中，无条件的从程序的一行跳转到另一行去执行，即将程序控制转移到指定的标签。

## 指令格式

**JMP LBL[i]**

指令结构如下：



示例：

1: JMP LBL[2]

## 14.4 子程序调用指令

### 说 明

子程序调用指令将程序控制转移到另一个程序（子程序）的第一行，并执行子程序。当子程序执行到程序结束指令（END）时，控制会迅速的返回到调用程序（主程序）中的子程序调用指令的下一条指令，继续向后执行。

需要选择子程序名，或直接新建一个子程序，详细操作请参照前面章节的操作说明。

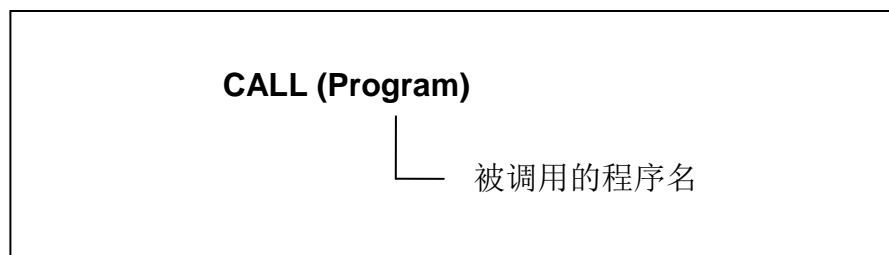
## 指令格式

## 14.流程控制指令

---

### **CALL (子程序名)**

指令结构如下：



示例：

1: CALL SUB1

## 15 其他指令

将上面未介绍的指令，归结为其他指令，这些指令包括：

- 坐标系指令；
- 用户报警指令；
- 倍率指令；
- 注释指令；
- 信息指令；
- 预读指令。

### 15.1 坐标系指令

坐标系指令用于改变机器人当前工作所使用的坐标系的设置。有以下两种坐标系指令：

- 坐标系设置指令—改变指定的坐标系的定义；
- 坐标系选择指令—改变当前选择的坐标系的序号。

#### 15.1.1 坐标系设置指令

##### 说 明

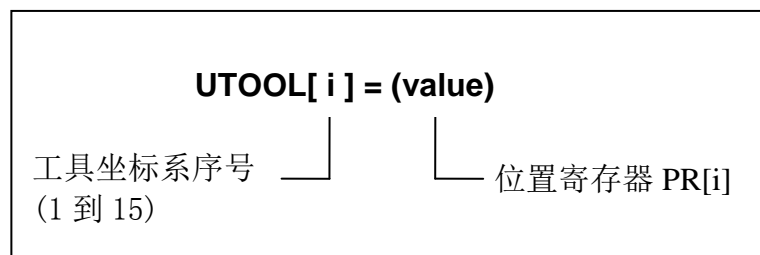
- 工具坐标系设置指令，改变由工具坐标系序号指定的工具坐标系的设置；
- 工件坐标系设置指令，改变由工件坐标系序号指定的工件坐标系的设置。

##### 指令格式

###### (1) 工具坐标系设置指令

**UTOOL[i]=(value)**

指令结构如下：



示例：

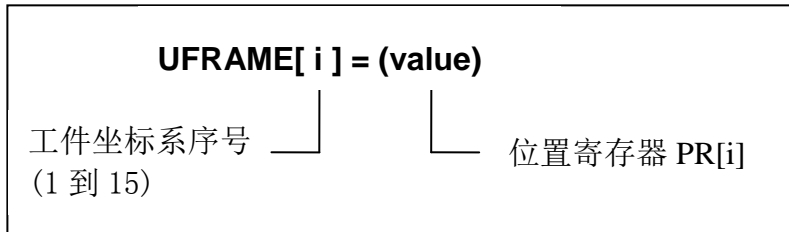
- 1: UTOOL[1] = PR[1]
- 2: UTOOL[R[2]] = PR[R[3]]

## 15.其他指令

### (2) 工件坐标系设置指令

**UFRAME[i]=(value)**

指令结构如下：



示例：

1: UFRAME[1] = PR[1]

2: UFRAME[R[2]] = PR[R[3]]

### 15.1.2坐标系选择指令

#### 说 明

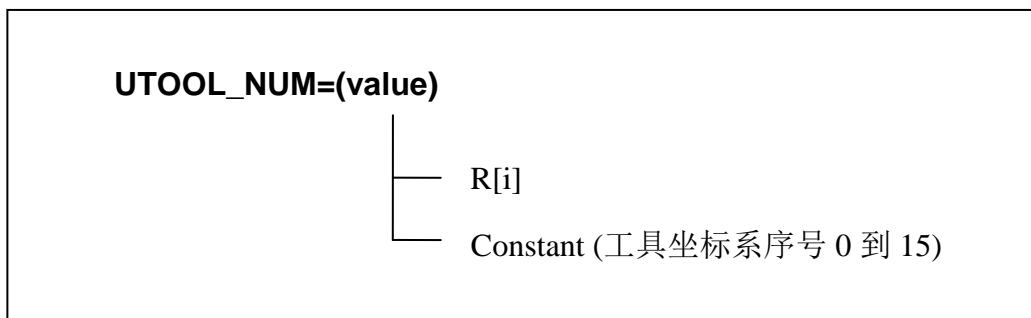
- 工具坐标系选择指令，改变当前工具坐标系的序号；
- 工件坐标系选择指令，改变当前工件坐标系的序号。

#### 指令格式

##### (1) 工具坐标系选择指令

**UTOOL\_NUM=(value)**

指令结构如下：



示例：

1: UTOOL\_NUM = 1

2: J P[1] 100% FINE

3: L P[2] 500mm/sec FINE

4: UTOOL\_NUM = R[2]

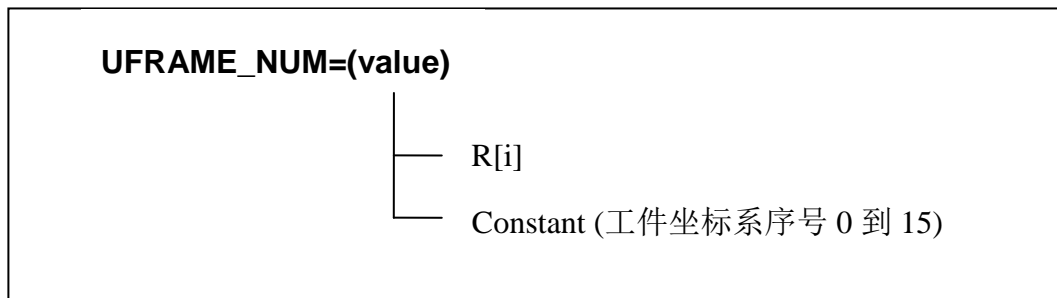
5: L P[3] 500mm/sec FINE

6: L P[4] 500mm/sec FINE

## (2) 工件坐标系选择指令

**UFRAME\_NUM=(value)**

指令结构如下:



示例:

- 1: UFRAME\_NUM = 1
- 2: J P[1] 100% FINE
- 3: L P[2] 500mm/sec FINE
- 4: UFRAME\_NUM = R[2]
- 5: L P[3] 500mm/sec FINE
- 6: L P[4] 500mm/sec FINE

## 15.2 用户报警指令

### 说 明

用户报警指令显示报警信息，并中止程序的运行。报警信息在系统信息的用户报警界面中设置。

### 指令格式

**UALM[i]**

指令结构如下:

## 15.其他指令

---

**UALM [ i ]**

└─ 报警号(9000 到 9099)

示例：

1: UALM[9004]

2: UALM[R[2]]

### 15.3 倍率指令

#### 说 明

倍率指令改变进给速度倍率。

#### 指令格式

**OVERRIDE = (value)%**

指令结构如下：

**OVERRIDE=(value)%**

└─ R[i]  
└─ Constant (1 到 200)

示例：

1: OVERRIDE = 100%

### 15.4 注释指令

#### 说 明

注释指令在程序中添加注释。注释不影响程序的执行。在注释指令中指定注释，一条注释最多可以包含32个字符。

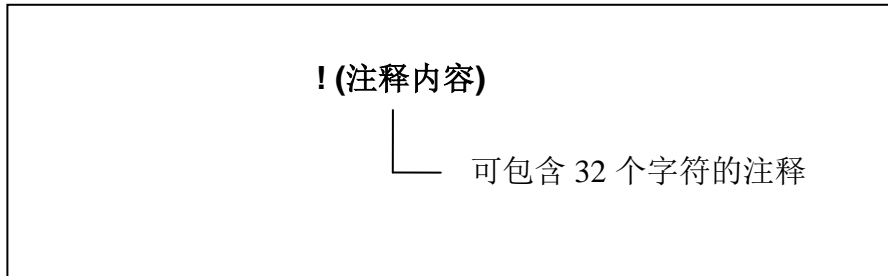
注释指令在程序选择输入界面中的表现形式是 **COMMENT**(注释)，选中并输入到

示教界面后，示教界面显示为“!”，详见前面章节的操作说明。

## 指令格式

**!(注释内容)**

指令结构如下：



示例：

1: ! APPROACH POSITION

## 15.5 信息指令

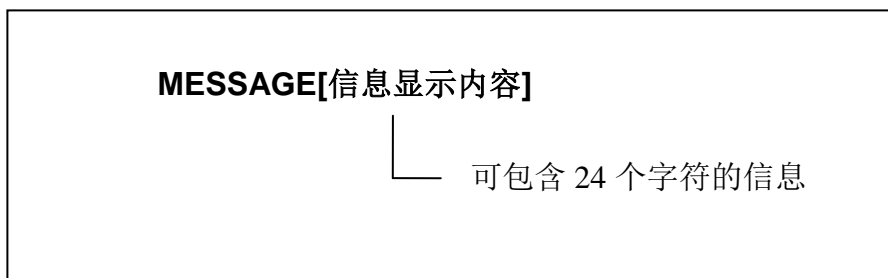
### 说 明

信息指令即在用户界面上显示指定的信息。一条信息最多可以包含24个字符。

## 指令格式

**MESSAGE [信息显示内容]**

指令结构如下：



示例：

1: MESSAGE[ DI[1] NOT INPUT ]

## 15.6 预读指令

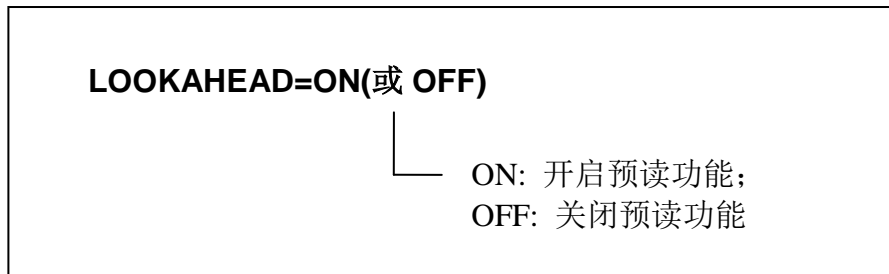
### 说 明

预读指令即设置开启或关闭超前预读功能。

### 指令格式

#### **LOOKAHEAD=ON/OFF**

指令结构如下：



示例：

1: LOOKAHEAD=OFF

超前预读功能通常配合运动指令中的定位路径方式 CNT 一起使用，当预读功能开启时，系统提前读取将要执行的程序行中的 CNT 指令，从而实现多点连续运动式的圆弧过渡。预读功能关闭后，CNT 方式不再生效，实际效果等同于 FINE 方式。

啊



## 16 附录：程序报警定义

报警号	报警级别	报警定义	备注
5100	2	分配内存失败	
5200	2	加载程序失败	
5300	2	语法匹配失败	
5400	2	指令格式错	
5500	2	位置变量(P)地址无效	位置变量(P)地址最大 999
5600	2	位置寄存器(PR)地址无效	位置寄存器(PR)地址最大 99
5700	2	寄存器(R)地址无效	寄存器(R)地址最大 199
5800	2	寄存器(AR)地址无效	参数寄存器(AR)地址最大 9
5900	2	缺少位置参数	移动指令(J、L、C)未指定位置参数
6000	2	缺少速度参数	移动指令(J、L、C)未指定速度参数
6100	2	缺少数据单位	缺少单位，如 mm/sec
6200	2	缺少轨迹过渡类型	移动指令(J、L、C)缺少过渡类型(FINE/CNT)
6300	2	跳转目标无效	
6400	2	子程序不存在	
6500	2	子程序嵌套层次过多	最多允许嵌套调用 10 层
6600	2	缺少执行语句	IF 指令缺少执行语句 JMP 或 CALL
6700	2	参数无效	CALL 指令传递的参数(AR)无效，仅支持字符串型和数型参数
6800	2	参数过多	CALL 指令最多可传递 10 个参数(AR)
6900	2	除零	
7000	2	数据类型不一致	位置寄存器指令操作数类型不一致，应统一为关节位置或直角坐标位置
7100	2	用户报警号无效	用户报警号范围 9000-9099，超出此范围无效
7200	2	数字 IO 地址无效	地址超出允许范围
7300	2	模拟量 IO 地址无效	地址超出允许范围